

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 004.89

Шаплыко
Илья Михайлович

Адаптивная модель обучаемого для системы управления
обучением в ВУЗе

ДИССЕРТАЦИЯ

на соискание степени магистра технических наук
по специальности 1-31 80 10 «Теоретические основы
информатики»

Научный руководитель
Захаров В. В.
кандидат технических наук,
доцент кафедры ИИТ

Минск 2020

СОДЕРЖАНИЕ

Перечень условных обозначений	3
Введение	4
1 Современные подходы к управлению процессом обучения в высших учебных заведениях	6
1.1 Модели высшего образования	6
1.2 Управление процессом обучения с использованием адаптивных обучающих систем	7
1.3 Анализ педагогических теорий обучения	10
1.4 Общепринятая модель обучаемого	11
1.5 Характеристики обучаемых, используемые в современных адаптивных системах	13
1.6 Оценка знаний обучаемого	16
1.7 Адаптационные модели	19
1.8 Анализ адаптивных обучающих систем и оболочек для их разработки	20
1.9 Выводы	23
1.10 Постановка задачи	23
2 Модель обучаемого	25
2.1 Математические модели пользователей в адаптивных обучающих системах	25
2.2 Разработка алгоритма составления учебного плана	29
2.3 Адаптация по времени обучения	34
2.4 Выводы	38
3 Реализация адаптивной модели обучаемого для системы управления обучаемого в ВУЗе	39
3.1 Разработка функциональной модели предметной области	39
3.2 Разработка укрупнённой архитектуры программного средства	42
3.3 Разработка базы данных программного средства	44
3.4 Выводы	46
Заключение	47
Список использованных источников	49

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ

- АОС — адаптивная обучающая система;
- ВУЗ — высшее учебное заведение;
- API — application programming interface (программный интерфейс приложения);
- CSS — Cascading Style Sheets (каскадные таблицы стилей);
- ES — ElasticSearch;
- GUI — Graphical User Interface (графический интерфейс пользователя);
- HAM — HyperText Abstract Machine (абстрактная машина гипертекста);
- HTML — HyperText Markup Language (язык гипертекстовой разметки);
- IMS — Intellectual Metasystem (интеллектуальная Метасистема);
- JSON — JavaScript Object Notation (нотация объектов JavaScript);
- OSTIS — Open Semantic Technology for Intelligent Systems (открытые семантические технологии для интеллектуальных систем);
- PDA — Personal Digital Assistant (личный цифровой секретарь, карманный персональный компьютер);
- UML — Unified Modeling Language (унифицированный язык моделирования);
- WWW — World Wide Web (всемирная паутина).

ВВЕДЕНИЕ

В условиях интенсивного развития техники и технологий существует потребность в эффективной передаче знаний, опыта, умений и навыков. Изменение структуры и содержания высшего профессионального образования, направленное на увеличение разнообразия высшего образования и разработку нового содержания, обеспечивающего социальное, экономическое и культурное развитие, формирует новые потребности в информации и определяет структуру современной информационной среды. Появление новых форм обучения, связанных с развитием мобильного и дистанционного образования, обусловило потребность в разработке обучающих систем и инструментальных средств для разработки таких систем.

В высшей школе есть большой потенциал для развития новых форм и методов обучения на основе новых информационных технологий: наличие соответствующей материальной базы, квалифицированных инженерно-педагогических и научно-педагогических кадров, оперативная разработка новых учебных планов и курсов, развитие коммерциализации в образовательной деятельности и т.п. С другой стороны, современные условия быстрого технологического развития диктуют новые подходы и методы поддержки высокого профессионального уровня руководящих и инженерных кадров. Непрерывный процесс повышения квалификации и обновления профессиональных знаний стал неотъемлемой частью работы менеджеров и технических работников всех уровней в большинстве крупных корпораций и банков мира, что является необходимым условием выживания коммерческой организации при существующей жесткой конкуренции.

В настоящее время формируется новый принцип построения обучающих систем: процесс обучения в них рассматривается как процесс управления знаниями обучаемого. В рамках этого подхода ведутся работы по разработке интеллектуальных систем управления обучением. Наиболее соответствующими принципам современного образования и перспективными с точки зрения поддержки процесса обучения являются адаптивные обучающие системы, т.е. системы, поддерживающие индивидуальный подход в обучении. Однако модели и методы, применяемые в системах, в недостаточной степени учитывают особенности уровня подготовки студентов в ВУЗ-ах. Поэтому актуальной является задача разработки принципов построения адаптивных обучающих систем, предназначенных для управления процессом обучения в технических вузах.

Центральным объектом интеллектуальных обучающих систем является модель обучаемого.

Цель создания модели обучаемого – построить процесс обучения таким образом, чтобы при минимальных затратах времени добиться максимальной эффективности обучения.

Наличие модели обучаемого позволит организовать гибкое (адаптивное) управление процессом обучения. Адаптивная обучающая система может содержать несколько вариантов изложения одного и того же материала. Решение о продолжении обучения по одному из вариантов должно приниматься на основании значений параметров модели обучаемого. Значения параметров могут также учитываться системой при выборе контрольных заданий, лабораторных работ и т.п.

Все это позволит учитывать индивидуальные особенности обучаемых, чтобы сделать процесс обучения для них более эффективным, более комфортным, а также повысить уровень знаний, умений, интерес к предмету у обучаемых и в значительной степени повысить эффективность и качество учебного процесса.

Цель и задачи работы. Целью данной магистерской диссертации является повышение эффективности управления процессом обучения за счет использования адаптивных обучающих систем.

Для достижения поставленной цели были выделены следующие задачи:

- анализ существующих моделей обучаемого;
- проектирование адаптивной модели обучаемого;
- разработка модуля адаптивной модели обучаемого;

Объектом исследования являются системы, обеспечивающие управление учебным процессом.

Предметом исследования являются модели, методы и средства разработки обучающих систем.

1 СОВРЕМЕННЫЕ ПОДХОДЫ К УПРАВЛЕНИЮ ПРОЦЕССОМ ОБУЧЕНИЯ В ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЯХ

1.1 Модели высшего образования

В статье Е.Н. Филлипова[1] под обучением понимается процесс передачи знаний от носителя (источника) к получателю или процесс формирования знаний на основе данных, поступающей на органы чувств. Обязательные элементы обучения – носитель знаний (контента), получатель знаний, знания (контент), способ движения знаний (метод, методика и каналы движения). Носителем знаний традиционно выступает тьютор, способ движения знаний основан при этом на взаимодействии обучаемого и тьютором. Появление новых форм обучения, обусловило расширение понятия носителя знаний, и обусловило потребность в разработке обучающих систем и инструментальных средств разработки таких систем.

Процесс управления знаниями можно представить в зависимости от вида отношений между тьютором и обучаемым. В статье Майка Фитзджеральда[2] выделены три модели высшего образования: ремесленная, промышленная и постиндустриальная. Первой моделью считается ремесленная модель, которая была основана на личных отношениях обучающегося с тьютором, и предполагала активное участие обучающегося. В дальнейшем в результате значительного расширения контингента высшего образования появилась промышленная модель. Следующий этап связан с развитием дистанционного образования и внедрением постиндустриальной модели, в которой реализуются принципы ремесленной модели обучения, но используются новые информационные технологии для обеспечения процесса обучения. Компьютеры при этом используются как средство для получения знаний из различных баз данных в локальных сетях и сети Интернет, и таким образом знания становятся доступны очень многим лицам. Однако анализ текущего положения дел в области внедрения новых информационных технологий показывает, что современные требования к приобретению знаний изменяются. Пользователи хотят иметь получать знания и формы их представления такими, чтобы это удовлетворяло каждого конкретного пользователя. Такой подход к получению знаний обусловил формирование четвертой модели, объединяющей достоинства дистанционного обучения и обучения «лицом-к-лицу». Будем называть такую модель «адаптивной» моделью. Модели высшего образования показаны на рисунке 1.1.

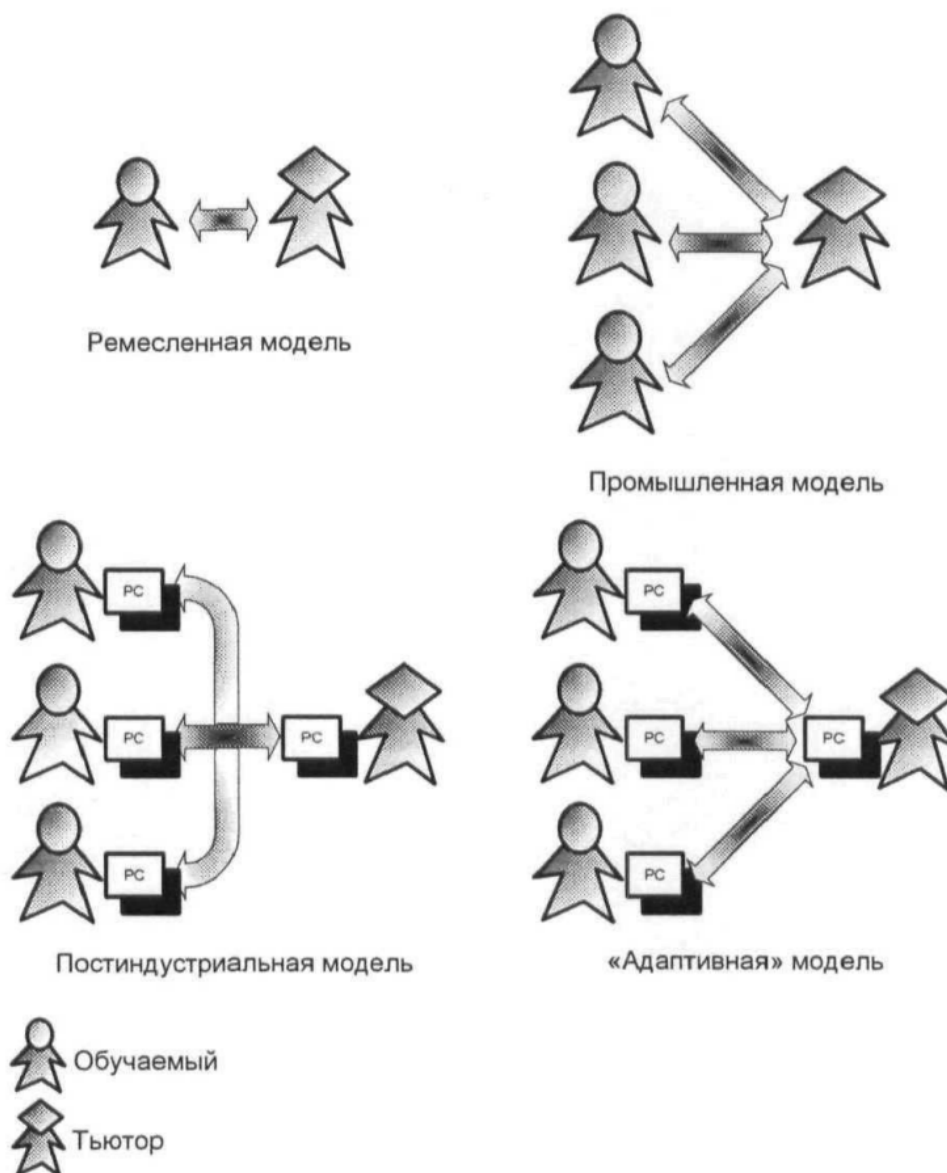


Рисунок 1.1 – Модели высшего образования

Адаптивная модель ориентирована на индивидуализацию массового высшего образования. Для внедрения такой модели необходимо использование новых информационных технологий, средств коммуникации и Интернета, а также новых подходов к обучающим системам, обеспечивающим индивидуализацию процесса обучения.

1.2 Управление процессом обучения с использованием адаптивных обучающих систем

Для управления процессом обучения на основе четвертой модели используются подходы, основанные на применении автоматизированных систем управления обучением. Формализация задачи обучения и критерия

обучения позволяет применить к ней методы оптимального и адаптивного управления. Такой подход к обучению коренным образом отличается от традиционных подходов. Он требует разработки новых для обучения категорий и понятий управления, и алгоритмов, реализующих принципы управления [3].

Существующие обучающие системы, используемые для управления процессом обучения, реализуют различные принципы управления. Известны три основных метода управления: программный, управление с обратной связью и адаптивный.

Применительно к рассматриваемой проблеме под программным управлением будем понимать управление с помощью обучающих систем, имеющих жесткую структуру и контент, то есть обучение в таких системах строится в соответствии с жестко заданной программой действий, независимо от реакций объекта управления и внешней среды. Управление с использованием принципа обратной связи предполагает возможность изменения структуры и содержания изучаемой предметной области, но изменения производятся извне системы, самим обучаемым или тьютором. И наконец, адаптивное управление подразумевает возможность системы самой настраиваться на конкретного обучаемого, на основе некоторой информации о нем и/или процессе обучения (рисунок 1.2).

Анализ тенденций исследований в области взаимодействия человек-компьютер показывает, что в настоящее время значительная часть разработок связана с созданием адаптивных систем, позволяющих не только обеспечивать пользователя информацией в любое время, в любом месте, в любой форме, но и предоставлять информацию, необходимую именно этому пользователю, и в удобном этому пользователю виде. При этом проектировщики таких систем сталкиваются с трудноразрешимой задачей создания программных систем для миллионов пользователей такими, как если бы они проектировались для одного отдельного пользователя. Особенно важна персонализация в обучении с помощью различных компьютерных обучающих систем, т.к. это позволяет обучающей системе в рамках одного курса предоставлять пользователю информацию в различном стиле и различного уровня сложности.

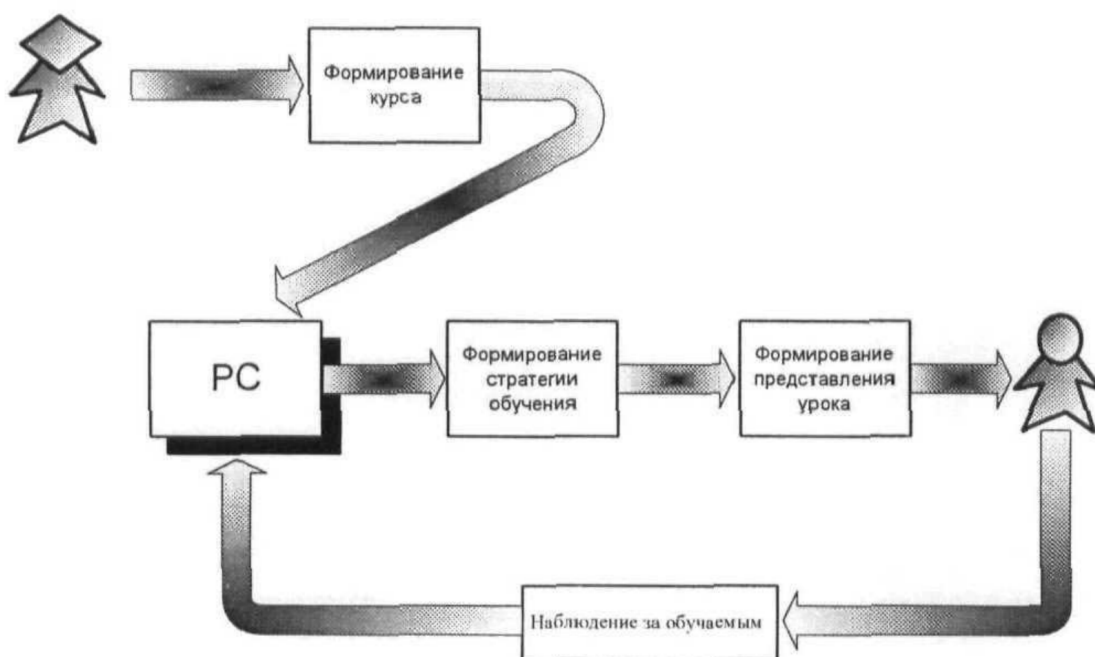


Рисунок 1.2 – Управление процессом обучения с использованием адаптивной модели.

Исследования в этой области разработки таких систем ведутся в трех направлениях: интеллектуальные обучающие системы, адаптивные гипермедийные системы, системы поддержки совместного обучения. Такие обучающие отличаются от обычных тем, что они используют интеллектуальные методики, которыми пользуются учителя в аудиториях и при индивидуальном обучении. Количество и разнообразие обучающих систем в WWW увеличивается изо дня в день. Но без адаптивного механизма, гипермедиа системы не способны предоставить, в большинстве случаев, учаемому необходимую информацию. Разработка адаптивных механизмов призвана решить две основные проблемы: ориентирование, в большом по объему информации, гиперпространстве, перегрузка лишней информацией. Большой вклад в область разработки методов обеспечения персонализации процесса обучения внесли П. Брусиловский, Бомон, Де Бра, А. Кобза, Г. Вебер, Фишер, Каплан, Кальви и другие [4–10]. Первые исследования в этой области относятся к работам Алена, Коена, Перо Элейн Рич [11–14]. В России исследования проблем адаптивного обучения отражены в работах [3, 15]. Адаптивные системы применяются в настоящее время в различных областях, где используются большие объемы информации и где эта информация может использоваться различными людьми с различными целями, знаниями и опытом.

П. Брусиловский [6] дает следующее определение понятию адаптивной системы: это гипертекстовая или гипермедийная система, которая отображает некоторые характеристики учаемого в виде его модели и использует эту модель для адаптации некоторых видимых аспектов системы к этому

обучаемому. Такая система должна удовлетворять следующим трем критериям: быть гипермедийной, иметь модель обучаемого, быть способной использовать эту модель для адаптации гипермедийной среды к этому обучаемому.

Адаптация в сетевых обучающих системах особенно эффективна. Во-первых, с такими системами работают пользователи, которые различаются по многим характеристикам. Приложения, спроектированные для одного класса пользователей, не будут подходить другим пользователям. Во-вторых, во многих случаях пользователь работает с системой самостоятельно, поэтому помощь, которую адаптивно предоставляют коллеги и учителя в обычном классном помещении, недоступна. В настоящее время обучение это одна из основных областей приложения адаптивных систем.

Таким образом, разработка систем, реализующих новые подходы к обучению, т.е. систем, способных «настраиваться» на конкретных пользователей, учитывать их индивидуальные особенности, является актуальной задачей.

1.3 Анализ педагогических теорий обучения

Анализ современных педагогических теорий обучения показывает, что наиболее широко используется конструктивистская теория обучения. Конструктивизм понимают как философский, эпистемологический и педагогический подход. В работе Чена [16] радикальный конструктивизм определяется как теория знания и познания и ее приложения к обучению. Конструктивизм определяется следующими принципами:

- знание не получается пассивно, ни с помощью восприятия, ни с помощью общения, а строится активно с помощью познания;
- функция познания адаптивна, она пытается увеличить свою адекватность и жизнеспособность.

В конструктивистской теории предполагается, что студенты должны обеспечиваться поддержкой при решении проблем, возникающих в реальной жизни. Обучающая среда должна моделировать контекст проблемы, в которой студент проявляет свою активность: он должен структурировать и разрешить проблему, собрать сопутствующий материал, выработать стратегии решения, т.е. решать реальные, жизненные проблемы. Для обеспечения этой модели конструктивистского обучения используется проектно-ориентированный подход.

В работах П.Я. Гальперина, В.Д. Шадринова, В.П. Беспалько, Н.Ф. Талызиной, Г.В. Суходольского [1, 17–19] рассматривается деятельностный подход к обучению. Важнейшей исходной посылкой данного подхода является психическая деятельность как, единая система умственных действий, процесс формирования которой возникает на основе материальных дей-

ствий. Фактически, это индивидуальный процесс присваивания социального опыта, результатом которого являются знания. Продолжением данной концепции можно считать разработки, связанные с алгоритмизацией обучения. Идеологами данного подхода можно считать В.П. Беспалько, Н.Ф. Талызину. Основопологающей является идея о возможности алгоритмизации процессов учения и обучения. Развитие этого подхода привело к программированному обучению и контролю знаний. Сама идея трансформировалась от жесткости и однозначного понимания алгоритмов, до гибкого и даже «эвристичного». Сформировалось представление о деятельности как функционировании педагогической системы и процессов, происходящих в ее подсистемах - учебной, педагогической, научно-воспитательной. В связи с этим учебную деятельность преподавателя можно представить как модель подготовки обучаемого к профессиональной работе. Проектирование такой модели должно исходить из ее сущностно-содержательной основы. И так как новые информационные технологии изменили не только средства, но во многом и методы работы профессионалов, то эти изменения должны найти отражение в конечных целях подготовки. Для повышения качества обучающихся, и, в частности, адаптивных обучающих систем необходимо учитывать современные педагогические подходы к обучению и использовать их при разработке таких систем.

1.4 Общепринятая модель обучаемого

К настоящему времени накоплен большой опыт, как в области теоретических разработок, так и в области разработки систем, реализующих принципы адаптации процесса обучения. Современные подходы к разработке адаптивных систем основаны на выделении трех составляющих: модель обучаемого, модель предметной области и адаптационная модель.

Понятие модели обучаемого выделилось из более общего понятия модели пользователя. Модель пользователя относится к классу информационных моделей, она содержит некоторую информацию о пользователе, и использует ее для адаптации процесса работы в системе к этому пользователю. Под моделью обучаемого понимают модель пользователя, предназначенную для адаптации процесса обучения и привязанную к модели предметной области. В работе Вебера[20] приводится следующее определение модели обучаемого – это совокупность набора характеристик обучаемого, измеряемых во время работы системы с обучаемым и определяющей степень усвоения им знаний по изучаемому предмету и методов (правил) обработки этого набора. В большой степени вопроса выбора определяется опытом разработчиков адаптивных систем, и, в какой то мере, назначением этих систем. Модель предметной области описывает структуру информационного пространства. Обычно эта модель представляется сетью, связывающей элементы информа-

ционного пространства. Адапционная модель определяет, как обновляется модель обучаемого и как на основе информации, получаемой из модели предметной области и модели обучаемого, и интерактивного взаимодействия с обучаемым происходит процесс адаптации. Адапционная модель содержит адапционный механизм, который определяет, как обновляется модель обучаемого в процессе обучения и как на основе информации из модели обучаемого происходит адаптация информационного пространства к этому обучаемому.

Выделяют следующие этапы проектирования адаптивных систем (рисунок 1.3)[10]:

- сбор данных об обучаемом;
- использование данных для построения и обновления модели обучаемого;
- применение модели обучаемого для обеспечения адаптации.

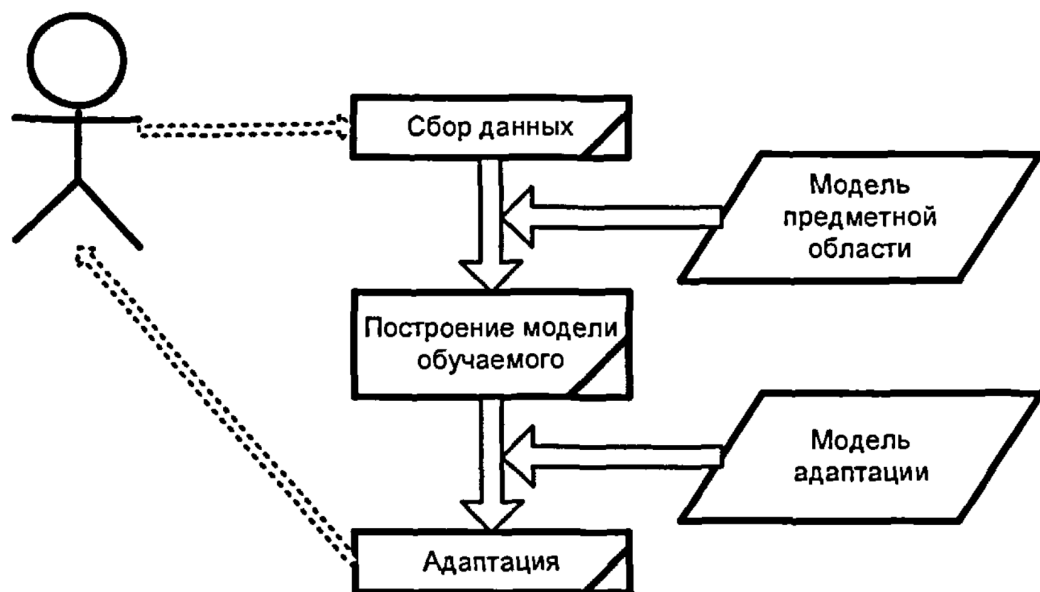


Рисунок 1.3 – Процесс адаптации в адаптивной системе.

В адаптивной системе должны быть автоматизированы все три этапа, то есть пользователь работает в системе, система наблюдает за ним, собирает данные о том, как идет процесс, использует эту информацию для построения модели пользователя, и обеспечивает адаптацию. Но систем, которые можно назвать адаптивными в этом смысле, практически нет. Выделяют несколько проблем, связанных с автоматизацией создания модели пользователя в адаптивных системах[9]. Основные две: во-первых, трудно обеспечить адекватность таких моделей, ошибка может быть допущена как при выводе модели, так и при адаптации (даже если модель сама по себе правильная), и во-вторых, некоторые компоненты модели вообще не могут быть дедуцированы и должны быть заданы непосредственно пользователем.

В то же время анализ показывает, что существуют достаточно эффективные методики автоматического построения модели. В [21] предлагаются такие методики, как трассировка действий пользователя и планирование знаний для определения целей.

Выбор способов сбора информации связан с назначением системы. Для обучающих систем автоматическое построение модели не может считаться лучшим вариантом. В обучающих системах наблюдение за действиями пользователя дает недостаточно информации для построения модели. Единственная информация, которую может хранить система, это путь и время, проведенное в каждом узле. Сам по себе путь и настройки навигации могут представлять интерес, но весьма сложно обновить модель по этой информации. Можно использовать такую информацию о навигации, как время, проведенное в каждом узле или количество посещений узла. Однако, тот факт, что пользователь посетил узел несколько раз или провел определенное время в этом узле, не гарантирует, что он прочитал внимательно содержание узла. Этот тип информации не достоверен и не может служить единственной информацией для идентификации обучаемого. Для обеспечения достоверности вывода об обновлении модели необходимо использовать дополнительные источники информации. Поэтому практически все адаптивные обучающие системы так или иначе используют внешние источники для получения информации о пользователе. Основным источником информации это информация, предоставляемая самим пользователем в различных формах. При этом специфической особенностью процесса моделирования является распределение обязанностей между пользователем и системой в процессе моделирования.

Большинство систем используют информацию об уровне знаний обучаемого и/или его интересы. Как правило, эта информация предоставляется самим обучаемым (цели) или вычисляется по результатам процесса обучения.

1.5 Характеристики обучаемых, используемые в современных адаптивных системах

В общем случае адаптивные системы могут адаптироваться к трем типам информации [2]: постоянные характеристики обучаемого (интересы, фон, стиль обучения, и т.д.), такие аспекты среды работы обучаемого, как место, время, используемое оборудование (компьютер, PDA, телефон) и сеть, информация об отношениях пользователя к концептам предметной области (уровень знаний и интересы.) Применительно к обучающим системам наиболее значимой и эффективной является адаптация к знаниям обучаемых. Также в таких системах могут принимать во внимание цели обучаемых, их интересы и ряд других характеристик. В большинстве суще-

ствующих адаптивных систем подразумевается, что все студенты имеют одну цель - изучить все концепты курса. В тех системах, которые поддерживают множественные цели обучения [22, 23], модель курса представляется атомарными структурами, что позволяет изменять содержание курса под индивидуальные цели обучаемого. При таком подходе к построению курса, цели обучаемого могут быть представлены множеством концептов, которые необходимо изучить. Цели обучения могут определяться как разработчиками курса, так и самим обучаемым. Такой механизм моделирования целей позволяет любому обучаемому добиваться тех целей, которые он определил. Наиболее естественным для обучаемого, при выборе целей является выбор тех концептов, которые будут им изучены при долговременном взаимодействии с системой. Система строит модель курса, основываясь на выбранных пользователем концептах, и также добавляются те концепты, без изучения которых невозможно перейти к выбранным концептам. Более усложненной формой представления целей обучаемого может быть их представление в виде пары значений: концепт - уровень сложности.

В модели, реализованной в системе InterBook [6], индивидуальные цели обучаемого представлены множеством последовательных целей. В [24] цели представляются организованным множеством в виде стека, позволяющим пользователю определить приоритетные для него цели на текущий момент, после чего данная цель помещается на вершину стека. В [25] реализован подход, позволяющий пользователю выделить основные цели обучения, которые затем декомпозируются на подцели. Последовательность элементарных целей, называемых слоями, представляет собой множество концептов, которые необходимо изучить. В [26] используется представление текущей цели в виде набора пар «цель - значение», где значение обычно является вероятностью того, что соответствующая пользователем цель является текущей целью пользователя. Применительно к обучающим системам целями могут быть изучение какого-то конкретного параграфа, или решение какой-то конкретной практической задачи, или изучение всего курса, или желание научиться решать практику и т.д. В некоторых системах разделяют локальные цели (низкого уровня), которые меняются часто, и глобальные (высокого уровня).

Под опытом подразумевается вся информация, связанная с предыдущим опытом пользователя вне изучаемого гипермедиа системой предмета, которая может быть ему полезной, и сильно повлиять на процесс обучения. Сюда входят и профессия пользователя, и опыт работ (обучения) в смежных областях.

Предпочтения отличаются от других свойств обучаемого тем, что самостоятельно системе очень трудно их определить. Обучаемому необходимо информировать систему о них напрямую или косвенно (при помощи обратной связи). Система может обобщить предпочтения и применять их

для адаптации в новых областях. Предпочтения могут быть связаны как с выбором информации, так и с представлением интерфейса. По разным причинам пользователь может предпочитать некоторые узлы и ссылки другим, а так же некоторые части страниц другим. Эти предпочтения могут быть постоянными или переменными, т.е. зависеть от темы, цели и т.д. Пользовательские предпочтения отличаются от других свойств пользователя по некоторым аспектам.

Индивидуальные признаки это название группы пользовательских особенностей, которые вместе определяют его индивидуальность, например, персональные факторы (т.е. интраверт/экстраверт), познавательные факторы, стили обучения. Как и пользовательские навыки, индивидуальные признаки постоянны для пользователя и не могут быть изменены вообще, или могут изменяться только с течением долгого периода. В отличие от навыков, индивидуальные признаки обычно получают не путем простого интервью, а специальными психологическими тестами. На данный момент существует ряд исследований, в которых рассматривается использование индивидуальных признаков для адаптации в некоторых предметных областях, в частности, в [27] осуществлена адаптация к стилю обучения. Учет индивидуальных признаков требует знания о связях между этими признаками и возможными установками интерфейса. В [24] предлагается подход «черного ящика», т.е. индивидуальные признаки моделируются, при этом используются несимвольные технологии. В [22] параметр, характеризующий тип мышления обучаемого, представлен двумя значениями: интуитивным и теоретически-методологическим мышлением. При этом для оценки типа мышления используется кортеж из двух двоичных цифр, где $(1, 0)$ характеризует интуитивное мышление, а кортеж $(0, 1)$ - теоретически-методологическое мышление. Учет индивидуальных особенностей позволяет реализовывать деятельностный подход к обучению. Таким образом, выбор характеристик, используемых для построения модели пользователя, определяется назначением системы, выбранными способами сбора информации, личным опытом разработчиков.

Результаты анализа характеристик, используемых в современных адаптивных системах, показаны в таблице 1.1.

Таблица 1.1 – Характеристики обучаемого, используемые в адаптивных системах

Система	Знания	Цели	Предпо- чтения	Фон	Опыт	Скорость изучения	Инд. хар-ка
АНА	+						
ELM-ART	+						
INTERBOOK	+	+			+		
KBS	+	+			+	+	
PT	+		+				
Нунecosum				+			
HYPERCASE		+					
HYPERFLEX		+					
UMT	+						
PUSH	+						
SYPROS	+						
SHIVA	+						
SYDNEY							+
INSPIRE							+

Таким образом, важным этапом разработки адаптивной системы является определение характеристик обучаемого, которые можно использовать для адаптации. В большинстве систем используются ограниченный набор характеристик, например, знания и цели, либо единственная характеристика – знания обучаемого в изучаемой предметной области. Проблемы использования различных характеристик связан с отсутствием единых подходов к способам представления информации об обучаемых, и механизмов оценки этих характеристик в модели обучаемого.

1.6 Оценка знаний обучаемого

Наиболее важной характеристикой пользователя является уровень его знаний в изучаемой предметной области. Адаптивная система, которая основана на знаниях пользователя, должна каким то образом распознавать изменения в уровне знаний и изменять соответственно модель пользователя.

В современных системах знания пользователя представляются оверлейной или стереотипной моделью (рисунок 1.4). Оверлейная модель основана на представлении предметной области в виде структурированной модели, состоящей из отдельных элементарных частей описания предметной области и представляет знания обучаемого как некий уровень покрытия предметной области. Для каждого элемента знаний индивидуальная модель обучаемого хранит некоторое значение, которое определяет уровень его знаний этого элемента. Знания пользователя показывают, насколько хорошо он знает

этот концепт. Оверлейные модели впервые были использованы в области систем интеллектуального преподавания и студенческого моделирования [21].

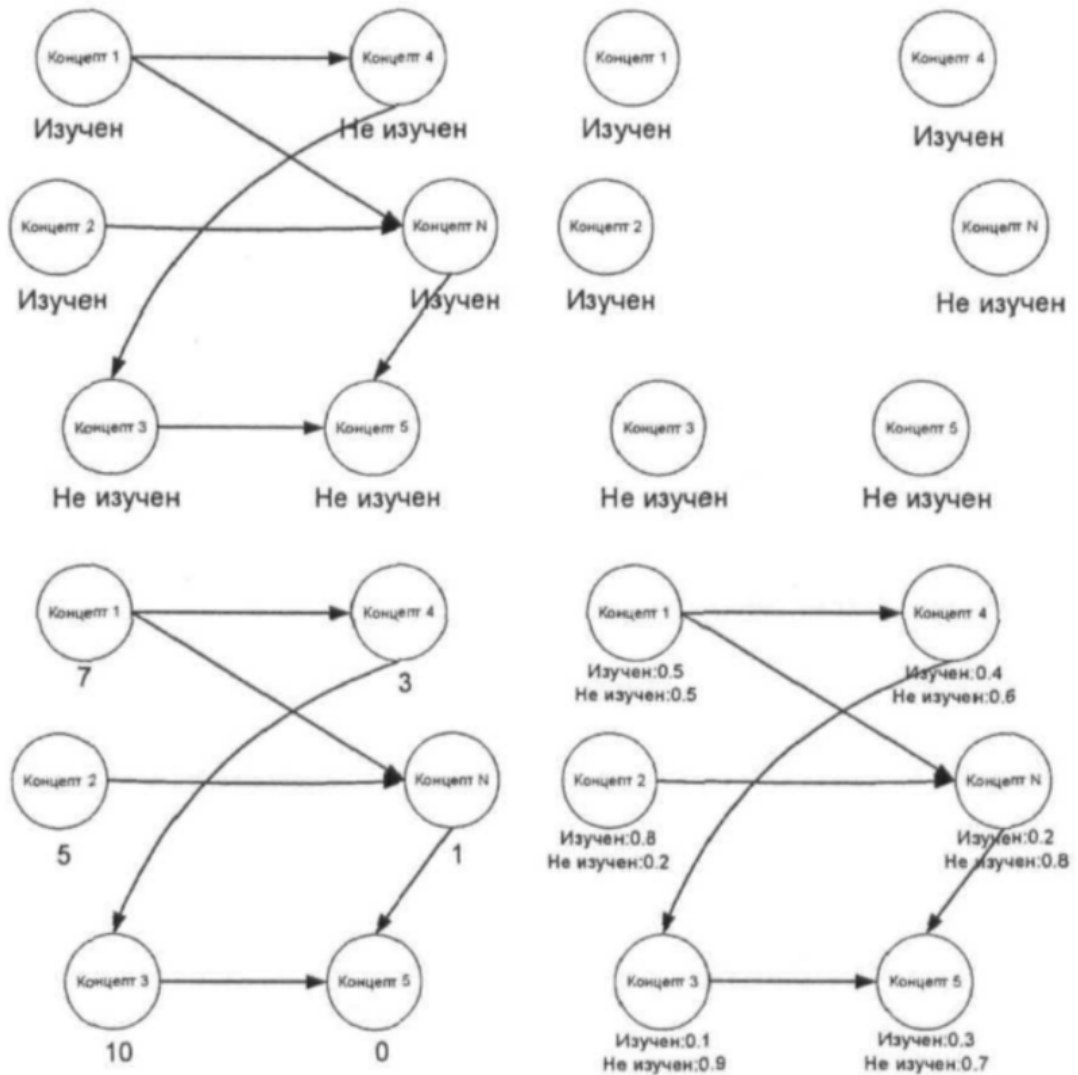


Рисунок 1.4 – Оверлейные модели пользователя.

К достоинствам таких моделей можно отнести их мощьность и гибкость, такие модели позволяют независимо оценивать знания пользователя по разным разделам. Но при выборе оверлейной модели не решена проблема инициализации пользователя – трудно определить начальные значения характеристик пользователя после короткого наблюдения за ним.

По способу оценки знаний пользователя оверлейные модели делят на: бинарные (оценка изучено/не изучено), взвешенные (количественная оценка), вероятностные (оценка рассчитывается на вероятностной шкале). В [7] используется Булева оверлейная модель. Возможно использование наборов значений, или процент или число в интервале (0, 1) [28]. В системе АНА! используется логфайл для каждого обучаемого, в котором отмечается

время посещения страницы и время ухода со страницы [8]. Все современные адаптивные системы обеспечивают фиксированный набор атрибутов.

Стереотипная модель [14] выделяет несколько типичных, «стереотипных» пользователей.

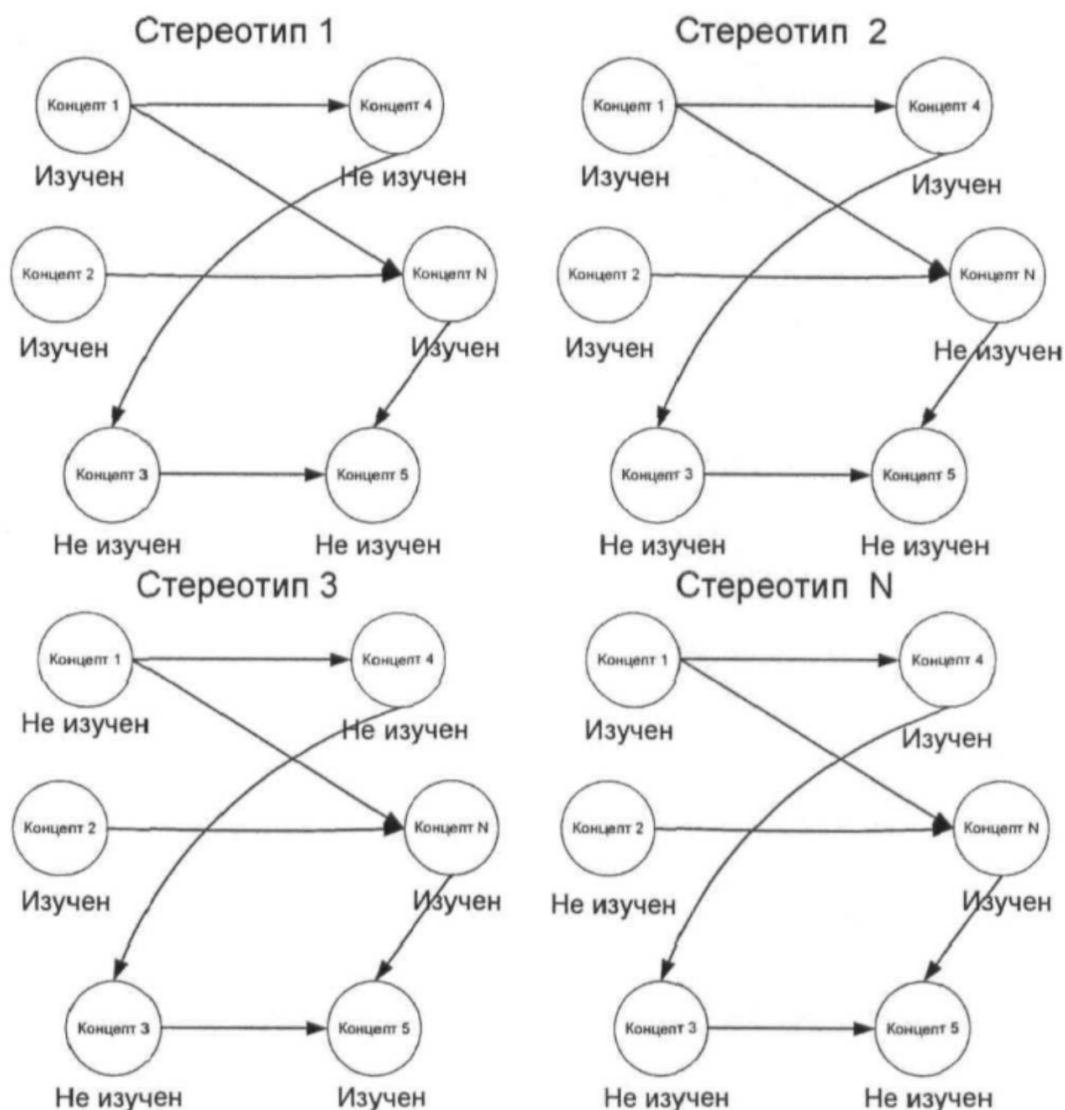


Рисунок 1.5 – Стереотипные модели пользователя.

Она также представляется как набор «стереотип – значение», где значением может быть не только «да» или «нет» (что означает, что пользователь принадлежит или не принадлежит этому стереотипу), но и вероятность принадлежности к стереотипу.

Стереотипная модель более простая и менее мощная, чем оверлейная модель, но ее легче инициализировать и управлять ею.

Анализ моделей обучаемого по признакам способов оценки знаний и учета зависимости обучения показан в таблице 1.2.

Таблица 1.2 – Анализ моделей обучаемого

Модель обучаемого	Тип оценки знаний в модели	Учет зависимостей изучения
Стереотипная	Количественный	–
Оверлейная	Булева переменная	+
Векторная оверлейная	Булева переменная	–
Взвешенная оверлейная	Количественная	+
Вероятностная оверлейная	Вероятностная	+

Можно выделить два основных подхода к построению моделей обучаемого. В экспертно-обучающих системах под моделью обучаемого понимают набор характеристик (параметров) и совокупность правил, которые на основании значений этих характеристик управляют процессом общения системы с обучаемым. В других классах обучающих систем под моделью обучаемого обычно понимают набор параметров, измеряемых во время работы системы с обучаемым и определяющей степень усвоения им знаний по изучаемому предмету.

Анализ показывает, что наиболее эффективной с точки зрения идентификации обучаемого является оверлейная модель. Можно управлять гибкостью такой модели с помощью выбора соответствующей шкалы оценки знаний обучаемого. Но для применения такой модели необходимо решить проблему инициализации такой модели.

1.7 Адаптационные модели

Для обеспечения адаптации информационного пространства необходимы способы отображения модели обучаемого необходимы механизмы. Понятие моделирования поведения адаптивной системы (моделирование адаптации) включает в себя определение логики адаптации (она отвечает за информацию доступную из модели и определяет необходимость адаптации) и определение действий адаптации (какие собственно действия должны быть произведены системой для достижения выбранной адаптации). Таким образом, адаптационная модель относится к классу поведенческих моделей. Логика адаптации основана на расчете текущих знаний пользователя по концепту предметной области. Существуют различные подходы к решению этой задачи. Как правило они основаны на правилах, причинно-следственных связях, т.е. всех способах, основанных на механизмах логического вывода. Определение логики адаптации зависит от выбранной модели обучаемого.

В [29] введено понятие эталонной модели, которая определяет способы отображения функциональности модели адаптации. Разработан ряд эталонных моделей, такие как НАМ[29], модель Треллиса[30], модель Декстера[31], эталонная модель Халаса и Шварца, формальная модель Ланга[32], более общая объектно-ориентированная модель Тауэр[7], модель АНАМ[33]. В этих моделях реализованы различные способы для отображения функциональности адаптационной модели. Однако эти модели описывают поведение системы на концептуальном уровне, при этом разработка конкретных механизмов адаптации в виде правил, остается задачей разработчика приложения. Кроме того, применение таких моделей требует знания соответствующих языков описания модели.

В [15] исследуются механизмы, основанные на нейронных сетях в применении к задаче адаптивного обучения. В такой постановке входной вектор представляет собой набор параметров модели обучаемого, а выходной вектор - код, определяющий методику обучения, соответствующую текущим значениям параметров модели обучаемого. По сути дела, этот подход использует стереотипное моделирование для выбора стратегии обучения для каждого стереотипа. В [34] для адаптации страниц используется механизм, основанный на метатегипертексте. Система Hupecosum [35] может дедуцировать уровень подготовки пользователя, основанный на использовании шаблонов навигационных путей пользователя. В системах Hupadapter, EPIAIM, ARIANNA, SETA используется адаптация, основанная на фреймовом представлении.

Анализ моделей и методов адаптации, применяемых в современных системах, показывает, что до настоящего времени не существует стандарта описания моделей АОС. В каждой системе реализуются свои подходы к обеспечению адаптации. Модели разработаны для реализации отдельных приемов адаптации, и не могут быть использованы для обеспечения других технологий и приемов адаптации.

1.8 Анализ адаптивных обучающих систем и оболочек для их разработки

В настоящее время существуют оболочки для разработки обучающих систем, которые широко используются для создания сетевых приложений. Наиболее известными являются TopClass (WBT Systems) и LearningSpace (Lotus), они представляют собой мощные обучающие системы, которые используются во многих университетах. С точки зрения определения адаптивной системы, TopClass можно определить как слабо адаптивную систему с возможностью адаптации связей. Система имеет возможность аннотации папок как «непрочитанные», если они содержат непосещенные страницы, или «новые», если они содержат новый материал. Однако, в этом случае

не используется структура предметной области, поэтому невозможно определить какая информация необходима для генерации страницы. Среда дистанционного обучения LearningSpace позволяет строить разветвленный курс – этот курс обладает дополнительными темами, на которые переходит обучаемый, в случае если по результатам проверки знаний он не может перейти на следующую тему. Система позволяет легко создавать и корректировать структуру курса, наполнять её материалами, подготовленными с помощью самых разных приложений. Структура курса и его материалы могут храниться на разных серверах в корпоративной сети или Интернет. Для работы в системе пользователю достаточно иметь на своём компьютере только браузер. Интегральная среда разработки и использования сетевых курсов WebCT [36] является одним из самых мощных и популярных в мире средств разработки и применения сетевых курсов. Технология сетевого обучения WebCT поддерживает стандарты IMS. В оболочке предоставляются возможности для создания Web-курсов, которые сложно создать разработчику-непрограммисту. WebCT использует браузер и графический интерфейс пользователя (GUI). Рассмотренные оболочки предоставляют широкие возможности для разработки электронных обучающих курсов, но они не позволяют обеспечивать адаптацию процесса обучения, то есть не имеют возможности моделирования пользователей.

Впервые понятие оболочки моделирования пользователя (shell) было введено Тимом Финнином. В [37] описана система моделирования пользователя GUMS, в которой использовалась иерархия стереотипов пользователей и система правил на Прологе. В системе используются подходы, основанные на сравнении фактов о пользователе, полученных системой, с текущим предположением, и ответах на запросы приложений относительно текущего предположения о пользователе. GUMS создала основу для функциональности общих (т.е. независимых от приложения) систем моделирования пользователя, а именно представления выбранных служб моделирования пользователя, которые одновременно могли быть настроены в время разработки. А.Кобза впервые использовал термин «оболочки моделирования пользователя» для обучающих систем[5]. В настоящее время работы ведутся в направлении создания оболочек, предназначенных для построения различных адаптивных обучающих курсов. Наиболее известными и используемыми являются такие системы, как АНА!, InterBook, KBS Hyperbook. InterBook – оболочка для создания адаптивных электронных учебников для Web[6]. Система основана на концептном подходе, предложенном П. Брусиловским [38] и развитом далее в системе для изучения языка LISP ELM-ART. В системе используются два типа знаний: знание об изучаемой предметной области и знание об обучаемом. Модель предметной области является основой для структурирования контента адаптивного электронного учебника. Модель адаптации основана на встроенных адаптационных правилах. Ос-

новой частью системы является онтология, предназначенная для описания предметных областей и проверки связности и полноты функций. В системе применяются три модели: модель предметной области, модель пользователя, и модель образовательных задач. Модель задач включает образовательные правила и онтологию (словарь, концепции и аксиомы), которые используются для сопровождения, тренировки и обучения, адаптированных к предпочтениям, статусу и уровню знаний конкретного обучаемого. Существующая версия находится в свободном доступе, но реализована только для компьютеров Apple Macintosh. В оболочке АНА!^[8] используется оверлейная модель, расширенная дополнительными концептами (которые не соотносятся с концептами модели предметной области), и включающая специальный псевдоконцепт, который содержит информацию о пользователе. Для построения курсов используются два редактора: графический редактор и редактор концептов. Модель адаптации использует адаптационные правила вида «если-то», автоматически генерируемые системой. Для вывода о состоянии пользователя используются его действия в процессе работы в системе и результаты тестирования. При этом достоверность вывода определяется разработанными правилами, интерпретирующими действия пользователя. Система не ориентирована на применение в обучении.

В системе KBS Hyperbook^[39, 40] используется модель студента, которая включает в себя: вектор знаний, который отображает знания разделов предметной области, модель зависимостей изучения между разделами, механизм вывода для оценки системой знаний пользователя этих разделов. В модели предметной области используются понятия базовых гипертекстовых абстракций, которые используются механизмом отображения системы. Абстракции хранятся в хранилище метаданных KBS Hyperbook. Система предоставляет набор средств для адаптации курса обучения в зависимости от целей и знаний конкретного обучаемого. Но при этом пользователь не инициализируется в начале процесса обучения. Информация об изменении уровня знаний поступает из внешнего источника.

В системе Sydney также реализованы некоторые принципы адаптивности. Это одна из немногих систем, которые позволяют учитывать индивидуальные характеристики обучаемого (его психологический тип). Каждому типу обучаемого сопоставляется своя траектория изучения учебного материала. По своим возможностям система может быть отнесена к слабо адаптивным системам.

В ^[41] индивидуализация обучения достигается возможностью изменений на когнитивном и презентационном уровне. Когнитивная составляющая определяет, как обучаемый знает предметную область. На уровне представления суть адаптации состоит в том, что каждый элемент предметной области может быть представлен несколькими способами, с разными уровнями понимания или сложности, и также в разных форматах.

Результаты анализа оболочек адаптивных систем показаны в таблице 1.3.

Таблица 1.3 – Анализ оболочек для разработки адаптивных обучающих систем

Система	Учет начального уровня знаний	Информационная поддержка	Учет цели обучения	Открытый доступ
Interbook	Нет	Нет	Да	Только для Apple Macintosh
АНА	Да	Нет	Нет	Да
KBS Hyperbook	Да	Нет	Да	Нет
KOD	Да	Нет	Нет	Нет
PAT	Нет	Нет	Нет	Нет

1.9 Выводы

В результате анализа моделей и методов адаптации, а также существующих оболочек для разработки адаптивных обучающих систем выявлены следующие проблемы:

1. Модели обучаемого, реализованные в этих оболочках, позволяют учитывать ограниченный набор характеристик обучаемого (знания и цели, или только знания);
2. Информация, необходимая для обновления модели обучаемого как правило должна поступать из внешних источников, т.е. не реализованы механизмы оценки знаний обучаемого с помощью адаптивных систем;
3. Механизмы адаптации не позволяют гибко интерпретировать изменение состояния обучаемого;
4. Не разработаны механизмы, которые бы позволяли обеспечить поддержку процесса обучения в случае недостаточной подготовки к изучению выбранных предметных курсов.

1.10 Постановка задачи

Проведен анализ проблем, связанных с обучением студентов в технических вузах. Опыт преподавателей различных дисциплин в техническом вузе показывает, что студенты испытывают трудности при изучении дисциплин, обусловленные следующими причинами:

1. разный уровень довузовской подготовки студентов, в школе или среднем специальном учебном заведении;

2. на старших курсах - плохое знание дисциплин, изучаемых на младших курсах;

3. для иностранных студентов, обучающихся в вузах, недостаточное понимание изучаемых дисциплин, особенно на младших курсах, из-за неуверенного владения русским языком;

4. недостаточное количество аудиторных часов для изучения дисциплин, требующих помощи со стороны преподавателя.

Поэтому для повышения эффективности управления процессом обучения в технических вузах необходимо использовать такие инструменты поддержки процесса обучения, как адаптивные обучающие системы.

На основании проведенного анализа для достижения поставленной цели были выделены задачи:

1. Провести анализ современных подходов к персонификации процесса обучения, систем, реализующих эти подходы, анализ предметных курсов, изучаемых в технических вузах, и определить критерии выбора характеристик обучаемого, существенных с точки зрения изучения этих курсов.

2. Разработать модель предметной области, позволяющую учитывать специфику предметных курсов и обеспечивать поддержку обучения за счет расширения информационными ресурсами из других предметных областей.

3. Разработать модель обучаемого, позволяющую учитывать индивидуальные характеристики, информацию об уровне подготовки в начале процесса обучения, и изменение уровня текущих знаний обучаемого в процессе обучения.

4. Разработать адаптационную модель и алгоритмы отображения электронного курса, позволяющие персонифицировать представление электронных обучающих средств на основе модели обучаемого, и реализовывать технологии высокоуровневой и низкоуровневой адаптации, необходимые для обеспечения требований к процессу управления обучением.

5. Реализовать предложенные модели и алгоритмы в системе для разработки адаптивных обучающих курсов.

2 МОДЕЛЬ ОБУЧАЕМОГО

2.1 Математические модели пользователей в адаптивных обучающих системах

В процессе обучения учитель использует специальные знания трех основных типов: знания о предмете обучения (чему учить), знания об ученике (кого учить), знания о стратегии и методах обучения (как учить). В современных интеллектуальных АОС указанные типы знаний явно выделены и представлены с помощью различных моделей, методов, алгоритмов и технологий искусственного интеллекта. На основе этих знаний АОС способна выполнять все основные функции учителя – предоставлять ученику учебный материал, контролировать степень усвоения учеником этого материала, помогать в решении задач, определять причины ошибок ученика и формировать на этой основе соответствующие учебные воздействия[42].

Процесс обучения в интеллектуальных системах сформулирован как задача управления. Ученик при этом выступает в качестве объекта управления (ОУ), а АОС выполняет функции устройства управления (УУ) – рисунок рисунок 2.1.

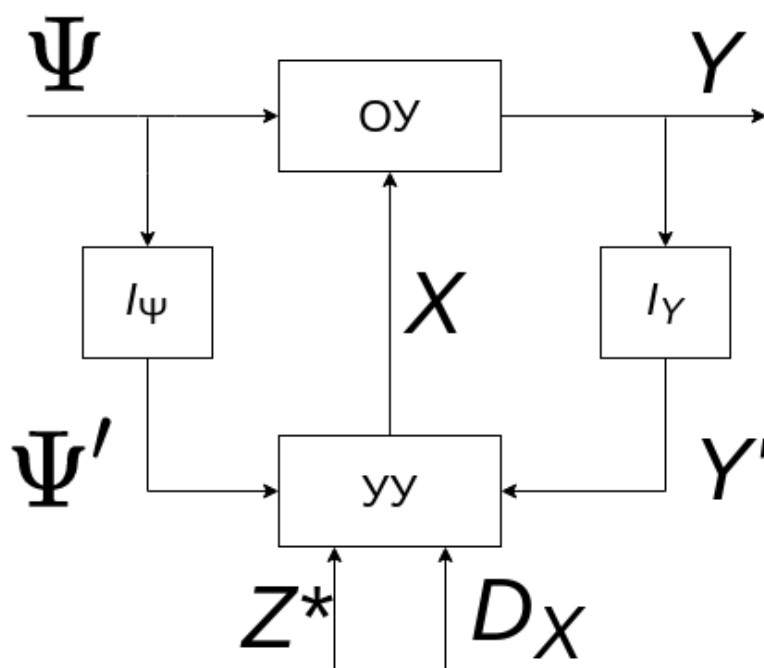


Рисунок 2.1 – Общая структура АОС

На рисунке приняты следующие обозначения: Ψ – состояние внешней среды; Y – состояние ученика; I_Ψ , I_Y – соответствующие измерители; Ψ' , Y' – результаты измерения величин Ψ , Y ; X – управляющие (обучающие и

контролирующие) воздействия; D_X – ресурсы (ограничения на управление); Z^* цель управления, состоящая в переводе ученика в требуемое состояние Y^* .

Во введенных обозначениях общее правило функционирования АОС представлено в следующем виде: АОС, получая на входе информацию о состоянии среды Ψ' и состоянии ученика Y' , а также информацию о цели Z^* и ресурсах D_X , выдает на выходе допустимое управление

$$X = A(\Psi', Y', Z^*) \in D_X$$

переводящее ученика из текущего состояния в состояние, близкое к Y^* . Здесь A – алгоритм управления процессом обучения.

Предполагая, что модель ученика, связывающая его наблюдаемые входы и выходы, имеет вид $Y' = ML(\Psi', X)$, задача синтеза оптимального управления X^* записывается в виде:

$$\text{Min} X \mu(Y - ML(\Psi', X)) = \mu(Y - ML(\Psi', X^*)), X \in D_X$$

где $\mu(*)$ – символ некоторой меры близости.

Цель управления Z^* записана как совокупность целей

$$\varphi_i = \varphi_i(\Psi', Y', X) = a_i,$$

$$\chi_j = \chi_j(\Psi', Y', X) \geq b_j,$$

$$\eta_k = \eta_k(\Psi', Y', X) \rightarrow \text{extr},$$

где $\varphi_i(\Psi', Y', X)$, $\chi_j(\Psi', Y', X)$, $\eta_k(\Psi', Y', X)$ – некоторые функциональные преобразования над переменными; a_i, b_i – заданные константы; $i, j, k = 1, 2, \dots$.

Целью $\varphi_i = a_i$ могут быть те знания, которые обязательно должны принадлежать множеству знаний ученика, $\chi_j \geq b_j$ – практический уровень знаний и умений, а значением $\eta_k \rightarrow \text{extr}$ может служить средний балл ученика по изучаемому предмету. Процесс синтеза модели ученика ML является повторяющимся (состоит из итераций), и в нем можно выделить два этапа: структурный и параметрический синтез.

На первом этапе в предлагается определить способы формализации величин Ψ', Y', X и зависимости $Y' = ML(\Psi', X)$. При этом величины Ψ', Y' задают в виде векторов

$$\Psi' = (\psi'_1, \psi'_2, \dots), Y' = (y'_1, y'_2, \dots),$$

а величину $X = (U, V)$ – в виде векторов обучающих $U = (u_1, u_2, \dots)$ и контролирующих $V = (v_1, v_2, \dots)$ воздействий. Зависимость $Y' =$

$ML(\Psi', X)$ представляют в виде некоторой функции $FL \in \mathbf{FL}$, определенный с точностью до вектора параметров $S = (s_1, s_2, \dots)$. Здесь \mathbf{FL} – некоторый класс функции.

Во введенных обозначениях модель ученика приобретает вид:

$$Y' = FL(\Psi', X, S), S \in D_S$$

где D_S – множество допустимых значений вектора параметров S .

На этапе синтеза параметров (или идентификации параметров) устанавливается значение элементов вектора S , т.е. значение параметров s_1, s_2, \dots . В этом случае могут использоваться три способа: идентификация в режиме нормальной эксплуатации объекта управления (при отсутствии специальных управляющих воздействий X), идентификация на основе организации специальных экспериментов с учеником, а также комбинированный подход. В общем виде задачу идентификации на основе специальных управляющих воздействий можно представить в виде

$$\begin{aligned} \min_s \|Y' - F_L(\Psi', X, S)\| &= \|Y' - F_L(\Psi', X, S^*)\| = \\ &= \|Y' - F_L * (\Psi', X, S)\|, S \in D_L, \end{aligned} \quad (2.1)$$

где $\| * \|$ – некоторая векторная форма, $F_L(\Psi', X)$ – необходимая модель ученика.

Для решения задачи (2.1) на данный момент существует несколько типов моделей ученика: оверлейные модели, имитационные модели, стереотипичные модели, разностные и пертурбационные модели.

Оверлейная модель (overlay model) описывает знания ученика, как некоторое подмножество всех знаний используемой модели знаний. Модель является частным случаем, так называемой, фиксирующей моделью ученика, простейшим вариантом которой является его скалярная модель, представляющая собой некоторую интегральную оценку знаний ученика в какой-либо шкале. Более сложными фиксирующими моделями ученика являются его векторная и сетевая оверлейные модели.

Векторная оверлейная модель представляет собой совокупность всех понятий рассматриваемого учебного курса и/или умений, соответствующих этому курсу, каждому из которых поставлено в соответствие значение «знает – не знает» или «умеет – не умеет». Преимуществом векторной модели является ее простота, а недостатком – то, что она не только не отражает когнитивные процессы и методы решения задач обучаемого, но и игнорирует связи между понятиями.

Сетевую оверлейную модель определяют как графовое представление семантической сети предметной области данного учебного курса, в котором каждому узлу и каждой дуге сопоставлена одна или несколько величин,

определяющих степень овладения учеником соответствующих понятий и отношений между ними. По способу оценки знаний ученика оверлейные модели разделяют на бинарные («изучено – не изучено»), взвешенные (используется та или иная количественная шкала), вероятностные (оценка рассчитывается по вероятностной шкале) и нечеткие (с использованием нечетких множеств).

Расширением сетевой оверлейной модели можно считать генетическую графовую модель. В отличие от оверлейной модели эта модель содержит не только нормативные понятия и отношения между ними, но и различные уточнения, обобщения, конкретизации и отклонения от нормативной модели, обусловленные особенностями знаний ученика. Например, с помощью отношения «уточнение» в генетической графовой модели могут быть описаны индивидуальные особенности понимания учеником тех или иных понятий.

Недостаток оверлейной модели ученика заключается в трудности ее инициализации. В этой связи в качестве модели знаний предметной области лучше использовать семантическую сеть, содержащую в качестве концептов обучающие элементы изучаемой предметной области, а также тесты для контроля уровня усвоения учениками этих элементов.

Оверлейную модель ученика формально можно представить в виде набора параметров $M(p_1, p_2, \dots, p_n)$, которые определяет педагог на стадии разработки курса. Значения параметров задаются арифметическими или логическими выражениями, которые включают операции, константы, другие параметры модели. Чтобы определить степень достижения учебной цели, текущую модель ученика можно сравнивать с экспертной (целевой). Последняя будет иметь те же параметры, но определенные «идеальными» значениями. Целевую модель можно обозначить $M_c(p_{1c}, p_{2c}, \dots, p_{nc})$. Чтобы модель ученика была более гибкой, можно допустить, что она содержит большее количество параметров, чем целевая. Это означает, что в процессе работы система или обучаемый могут добавить любое количество параметров по необходимости [43].

Модель разностного типа (дифференциальная модель) строят на основе различий между ответами ученика и соответствующими знаниями из базы знаний АОС. Важно, что модель данного типа позволяет учитывать не только отсутствие знаний у ученика, но и искажения этих знаний. С формальной точки зрения, разностную модель можно считать модификацией модели оверлейного типа.

Модель пертурбационного типа основана на предположении, что знания ученика и знания в базе знаний АОС могут частично не совпадать. Важной функцией модели пертурбационного типа является определение причин указанного расхождения знаний.

Имитационные модели предполагают представление знаний ученика в виде некоторых структур данных, а его умений – в виде процедур и

механизмов их интерпретации. Имитационная модель ученика, как правило, включает в себя его модели ошибок, ограничений и фальшправил.

Модель ошибок в той или иной форме фиксирует ошибки ученика, совершенные им при тестировании. Хотя модель ошибок традиционно относят к текущей модели ученика, эту модель проектируют заранее и ее, вероятно, следовало бы классифицировать, как нормативную модель ошибок. Ученик в процессе обучения совершает некоторые из ошибок этой модели, перечень которых составляет модель ошибок данного ученика.

Модель ограничений представляет собой совокупность нижних и верхних границ для уровней учебных достижений ученика. Здесь нижняя граница соответствует наиболее конкретным формам понятия или умения, а верхняя граница – их наиболее общим формам.

Основная идея стереотипичных моделей (stereotype models) состоит в выделении некоторого набора типовых (стереотипичных) по своим параметрам учеников. На основе этого набора для данного ученика определяют вероятность принадлежности его к каждому из указанных стереотипов. В качестве целевого выбирают стереотип, вероятность принадлежности к которому максимальна. По сравнению с оверлейной моделью данная модель значительно менее мощна, но ею легче управлять и, главное, инициализировать.

2.2 Разработка алгоритма составления учебного плана

Учебный план представляет собой совокупность дисциплин, покрывающих заданный набор компетенций, и формируется поэтапно. Для анализа ситуации и принятия решений необходима специальная процедура, которая может быть реализована в виде продукционной экспертной системы. Самое важное ограничение – это обеспечить компетенции, поэтому совокупность дисциплин и набор компетенций можно представить в виде двудольного графа, одна доля которого – дисциплины (для общеобразовательных программ высшего образования это дисциплины, относящиеся к вариативной части), а другая – компетенции. Каждой дуге соответствует вес, определяющий значимость дисциплины для данной компетенции. На этом этапе задача составления учебного плана представляет собой задачу о назначении в двудольном графе и заключается в поиске оптимального покрытия заданных компетенций. Оптимальным является покрытие, для которого затраты на изучение дисциплин будут минимальны и все компетенции обеспечены на 100%. Суммарная продолжительность учебных дисциплин, включенных в план, не должна превышать максимальный объем времени, отведенного на изучение этих дисциплин.

Многие дисциплины, включаемые в учебный план, находятся в зависимости от других дисциплин. То есть для изучения того или иного материала

требуются знания, которые обучающийся должен получить в процессе изучения другой дисциплины. В таком случае необходимо включать в учебный план не только искомую дисциплину, но и базовую дисциплину. Зависимость между дисциплинами можно представить в виде ориентированного графа D_G (рисунок 2.2), вершинами графа являются дисциплины, а дуги определяют зависимость между дисциплинами [44].

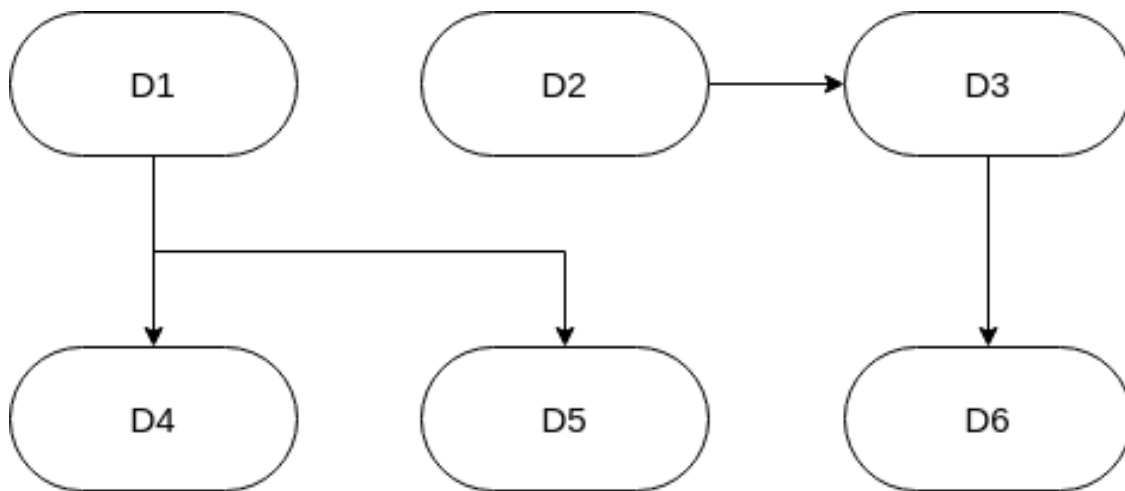


Рисунок 2.2 – Зависимость между дисциплинами

Данному графу присущи следующие свойства:

1. Существует начальная вершина, которая соответствует началу обучения, и конечная вершина, которая соответствует окончанию обучения.

2. D_G является бесконтурным графом, поскольку, освоив последовательность дисциплин, началом которой является некоторая дисциплина D' , обучающийся не может приступить к ее изучению снова.

3. Поскольку граф D_G является бесконтурным, то он может быть разложен на уровни, то есть представлен в виде иерархии (рисунок 2.3). При этом на верхнем уровне окажется фиктивная начальная вершина, а на нижнем фиктивная конечная вершина, которая соответствует окончанию процесса обучения. Конечная вершина, определяется на основе анализа компетенций, которыми должен обладать обучающийся по окончании обучения.

Если в качестве примера за начальную дисциплину взять вершину D_1 , а за конечную D_9 , то, как видно из рисунка 2.3, существует множество путей из начальной вершину в конечную. Из множества путей необходимо выбрать оптимальные для каждого обучающегося. Применение данного алгоритма позволяет использовать разложенные на уровни дисциплины в разрабатываемой процедуре составления учебного плана.

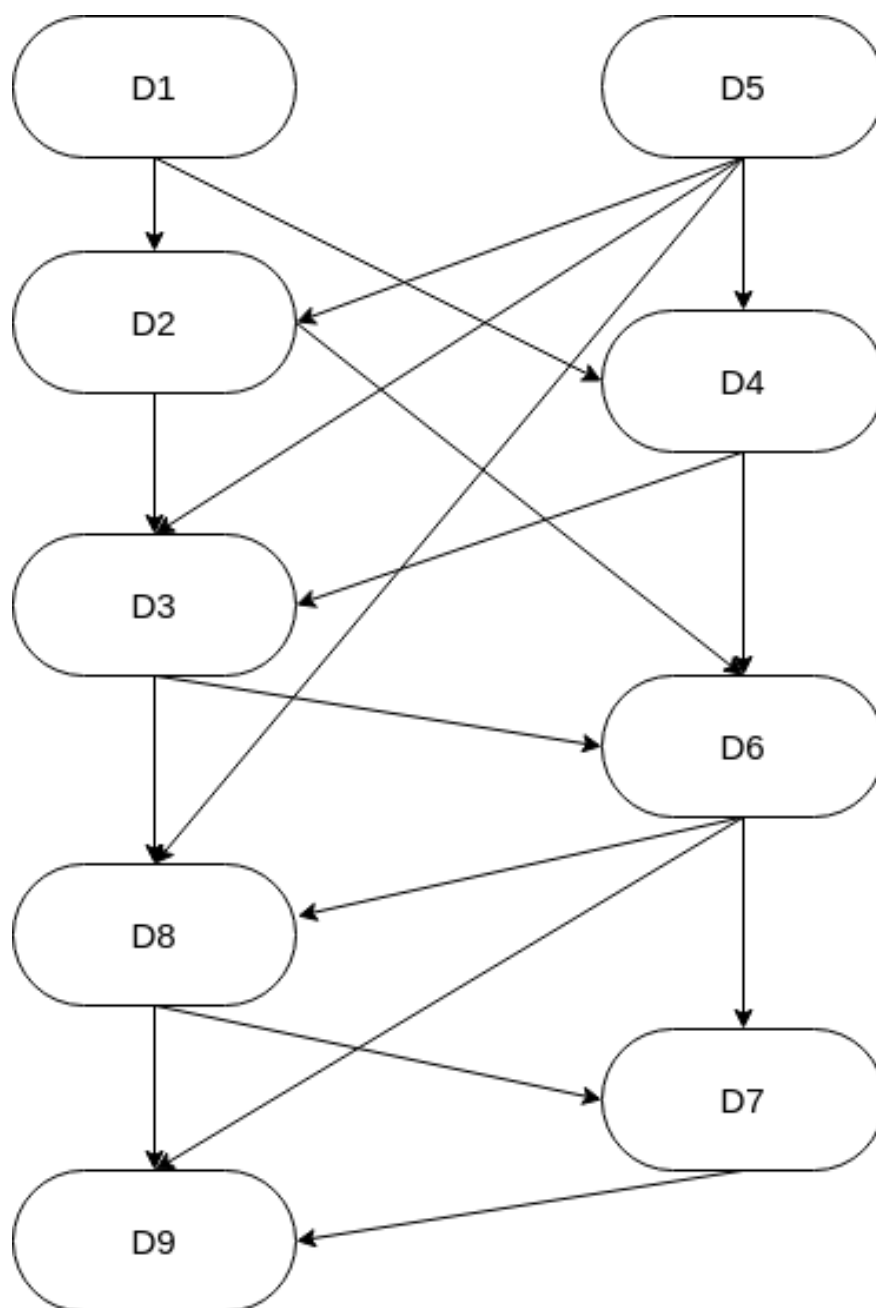


Рисунок 2.3 – Граф представленный иерархично

Однако, в процессе составления учебного плана необходимо учитывать то, что обучающийся мог ранее изучать некоторые из дисциплин. Для формирования совокупности дисциплин, которые будут включены в учебный план, разработан следующий алгоритм (рисунок 2.4):

1. определяем набор дисциплин для обучения;
 2. определяем матрицу эффективности;
 3. определяем главную дисциплину;
 4. определяем зависимые дисциплины;
 5. проверяем изучал ли обучающийся предшествующую дисциплину.
- Если данная дисциплина не изучалась, то добавляем ее в учебный план;
6. добавляем текущую дисциплину в учебный план;

7. проверяем степень покрытия каждой из компетенций;
8. если не все дисциплины просмотрены, то переходим на следующую дисциплину и далее на шаг 4, иначе построение учебного плана невозможно;
9. анализируем сформированный учебный план.

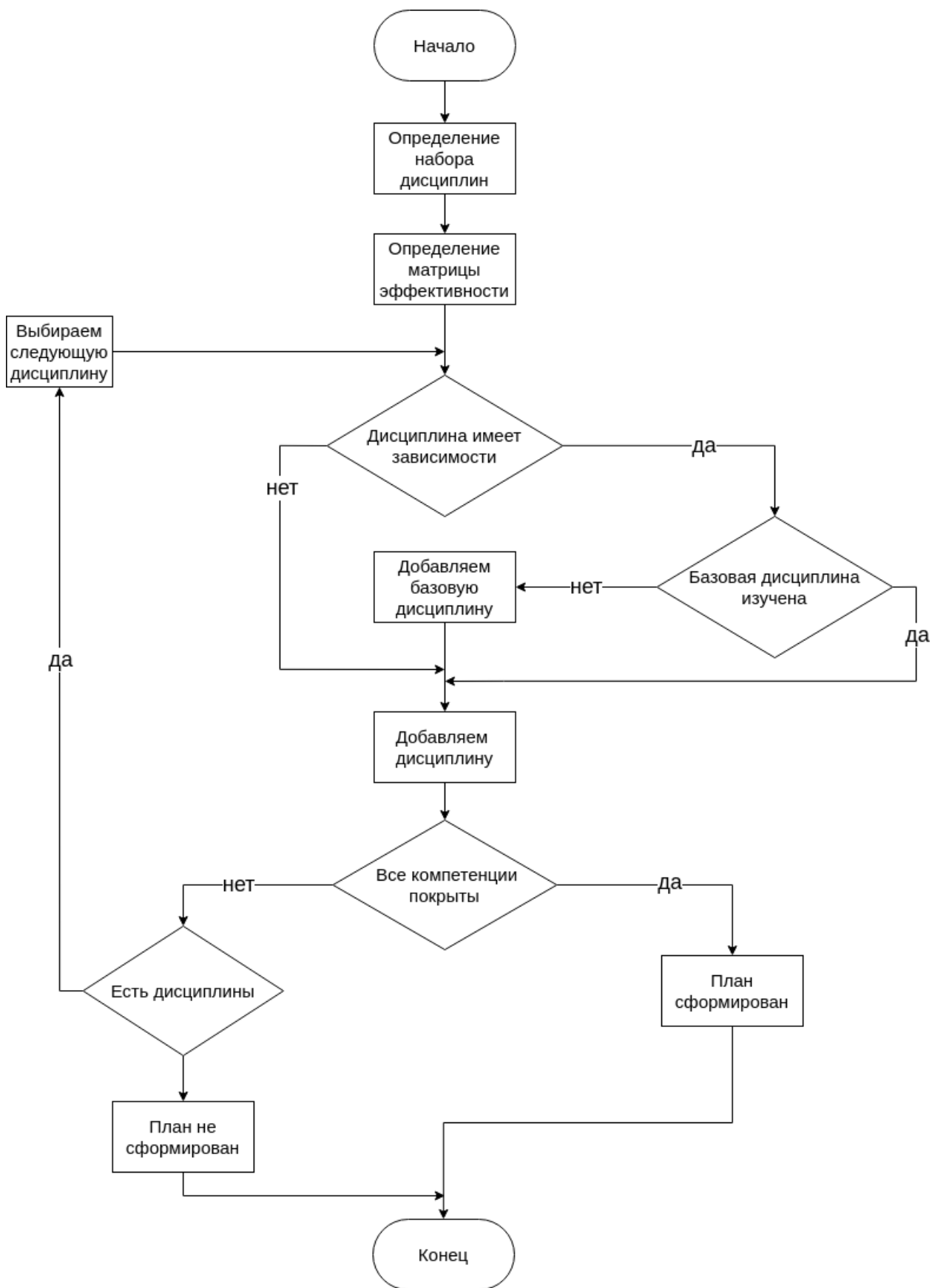


Рисунок 2.4 – Блок-схема алгоритма составления учебного плана

2.3 Адаптация по времени обучения

Отличительной чертой АОС является персонализация процесса обучения под особенности и требования каждого обучающегося. Адаптация по времени обучения является очень важным инструментом в процессе повышения качества обучения. Каждый обучающийся может принять участие не только в изначальной разработке учебного плана, но и в дальнейшей его корректировке.

Методы сетевого планирования широко применяются в учебном процессе. Они позволяют распределить учебные дисциплины во времени, спланировать загруженность преподавателей и учебных аудиторий. В разрабатываемой АОС методы сетевого планирования используются для внесения изменений в учебный план в процессе обучения. Данный подход позволит реализовать адаптивность не только на этапе планирования обучения, но и в его процессе.

Взаимосвязь между дисциплинами предполагает, что для изучения одних дисциплин нужно освоить некоторые другие, предшествующие им. Как сказано ранее, каждая дисциплина имеет время, необходимое для ее изучения. В процессе построения учебного плана необходимо определить этапы обучения, исходя из необходимого набора дисциплин, их взаимосвязи, времени, необходимого для их изучения и времени, отведенного на обучение. Для данной задачи перспективным направлением является применение методов сетевого планирования, совокупность которых образует эффективный инструмент управления проектами. Под проектом понимается комплекс работ, упорядоченных по времени отношением предшествования. Учебный план представляет собой совокупность дисциплин, которые необходимо изучать в определенной последовательности, поэтому в качестве его модели можно использовать сетевую модель, или сетевой график, который представляет собой ориентированный взвешенный бесконтурный граф. Каждой дуге графа поставим в соответствие дисциплину, продолжительность ее изучения – это вес дуги. Вершинам соответствуют события и определяют момент начала или завершения изучения каких-то дисциплин.

Сетевой график учебного плана для индивидуального обучающегося имеет следующие особенности:

1. существует начальная вершина, соответствующая началу обучения;
2. существует конечная вершина, соответствующая итоговому контролю знаний;
3. сетевой график является бесконтурным графом, иначе возвращаемся к ситуации, когда некоторую дисциплину и следующие за ней нужно изучать еще раз;
4. вес дуги – продолжительность изучения дисциплины, поэтому, если предположить, что она задается приближенно, то целесообразно задать:

- T_{\min} – минимальное время, отведенное на изучение данной дисциплины;
- T_{\max} – максимальное время, отведенное на изучение данной дисциплины;
- T_p – наиболее вероятное время, используемое в расчетах сетевого графика.

Если время задается одним числом (то есть определяется точно), то используется детерминированная модель (классический сетевой график). Однако этот параметр можно задать приближенно, и тогда целесообразно использовать модели, ориентированные на приближенный тип информации, а именно нечеткие сетевые модели или вероятностные сетевые модели.

Так как разрабатываемая система является адаптивной и предполагает индивидуализацию процесса обучения, обучающийся имеет возможность определить приоритетные дисциплины и уделить больше времени их изучению, в свою очередь, сократив время изучения остальных дисциплин. Главным условием выбора параметра T_p является $T_{\min} \leq T_p \leq T_{\max}$.

На начальном этапе сетевого планирования необходимо построить сетевой график, используя в качестве исходных данных набор дисциплин, определенный на предыдущем этапе.



Рисунок 2.5 – Обозначения вершин сетевого графа

В качестве примера приведем сетевой график, изображенный на рисунке 2.6.

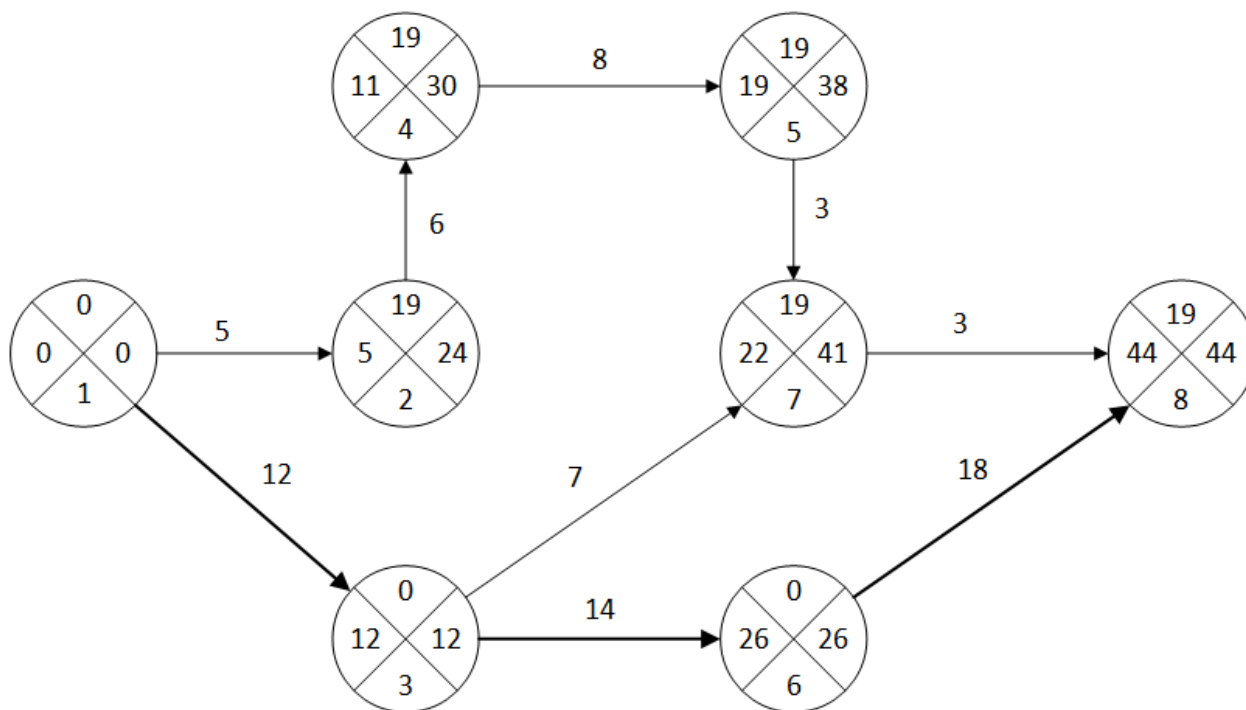


Рисунок 2.6 – Блок-схема алгоритма составления учебного плана

Каждой дуге-дисциплине соответствует время T_p (в учебных днях), отведенное на ее выполнение. Множество дисциплин – это множество работ (операций), которые необходимо выполнить в процессе обучения. Каждая вершина сетевого графика обозначает событие, которое соответствует началу или окончанию некоторой работы или совокупности работ. Поскольку сетевой график – бесконтурный граф, то он допускает разложение на уровни, т.е. представления в виде иерархии, которое характеризуется следующими особенностями:

1. вершины, находящиеся на верхнем уровне, соответствуют началу реализации учебного процесса (это может быть одна дисциплина, дающая базовые знания для остальных, или несколько дисциплин);;
2. из вершин данного уровня дуги ведут только в вершины нижеследующих уровней, определяя последовательность в изучении дисциплин;
3. уровни иерархии можно рассматривать как этапы учебного процесса, которые могут включать точки контроля знаний;
4. в иерархическом представлении можно выделить пути, ведущие из вершин верхнего уровня в вершины нижнего уровня, это позволяет выделить последовательности изучения некоторого подмножества дисциплин.

Основными временными параметрами сетевой модели являются:

- ранний срок наступления события (самый ранний из возможных сроков наступления события при условии выполнения всех предшествующих работ);
- критическое время (ранее время наступления завершающего события сети, которое определяет время, необходимое для изучения всех

дисциплин, включенных в учебный план;

– поздний срок наступления события (самый поздний из допустимых сроков наступления события, при котором не увеличивается общее время для изучения всех дисциплин, т.е. критическое время);

– полный резерв времени работы (время, на которое может увеличиться время освоения некоторой дисциплины, и при этом общее (критическое) время не изменится);

– свободный резерв времени работы (время, на которое можно увеличить продолжительность изучения дисциплины, не увеличив раннее время наступления события, которое соответствует началу изучения следующей дисциплины).

Критический путь – путь максимальной длины из начального события в завершающее, длина этого пути есть критическое время. Дуги (работы) и вершины (события), принадлежащие критическому пути, называются критическими.

Пусть t_{ij} – продолжительность работы (i, j) , N – номер завершающего события сети. Прежде чем определять параметры, необходимо осуществить топологическую сортировку сети, результатом которой является монотонная нумерация вершин, которая формирует список для просмотра вершин при определении временных параметров. В классическом случае временные параметры сетевой модели рассчитываются следующим образом:

1. Определение ранних сроков наступления событий. Положить $T_p(0) = 0$. Двигаясь по сети в порядке возрастания номеров вершин, определить $T_p(j)$ – ранний срок наступления события j по формуле 2.2.

$$T_p(j) = \max T_p(i) + t_{ij} \quad (2.2)$$

2. Определение критического времени по формуле 2.3.

$$T = T_N(j) \quad (2.3)$$

3. Определение поздних сроков наступления событий. Положить и двигаясь по сети в порядке убывания номеров вершин, определить – поздний срок наступления события i по формуле 2.4.

$$T_n(i) = \min T_n(j) + t_{ij} \quad (2.4)$$

4. Двигаясь от завершающего события к начальному, каждый раз выделять такую дугу (i, j) , для которой выполняется условие 2.5.

$$T_p(j) - T_n(i) = t_{ij} \quad (2.5)$$

5. Для каждой работы определить резервы. Полный резерв времени работы (i, j) – максимальное время, на которое можно увеличить продолжительность работы (i, j) , не изменяя продолжительность критического пути.

Рассчитывается по формуле 2.6.

$$R_{\Pi}(i, j) = T_{\Pi}(j) - T_p(i) - t_{ij} \quad (2.6)$$

Свободный резерв времени работы (i, j) – максимальное время, на которое можно увеличить продолжительность работы (i, j) , или отсрочить ее начало, не изменяя ранних сроков начала последующих работ. Рассчитывается по формуле 2.7.

$$R_c(i, j) = T_p(j) - T_p(i) - t_{ij} \quad (2.7)$$

Такая форма представления учебного плана позволяет в каждый момент времени определить, какие дисциплины изучаются, а анализ резервов позволяет изменять время, отводимое для изучения, не изменяя общего времени, отводимого для изучения учебного плана. Как сказано выше, дисциплины, расположенные на критическом пути, не имеют резерва времени на изучение. Из этого следует, что в случае задержек в процессе изучения данных дисциплин произойдет увеличение суммарного времени изучения учебного плана. Однако, данные издержки можно покрыть за счет уменьшения параметра T_p .

Описанный подход, опирающийся на применении методов сетевого планирования для составления учебного плана, позволяет максимально адаптировать процесс обучения под индивидуальные особенности обучающегося путем составления адаптированного учебного плана. Изменяемые временные параметры позволяют в случае необходимости скорректировать учебный план без увеличения суммарного времени. Данная особенность предлагаемого подхода позволяет системе работать с обучающимся в интерактивном режиме и подстраиваться под его возможности.

2.4 Выводы

1. Разработана математическая модель пользователей в адаптивных обучающих системах. Благодаря модели, система может предложить пользователю индивидуальный способ изучения курса, выполнить оптимальный подбор упражнений, активно поддерживать интерес к предмету на протяжении всего обучения.

2. Разработан алгоритм формирования учебного плана, который позволит покрыть все компетенции.

3. Предложен подход к организации учебного процесса (составлению календарного плана) с применением методов сетевого моделирования, при этом для каждой дисциплины определяются резервы времени, которые можно использовать либо для увеличения продолжительности ее изучения, либо для организации контроля знаний.

3 РЕАЛИЗАЦИЯ АДАПТИВНОЙ МОДЕЛИ ОБУЧАЕМОГО ДЛЯ СИСТЕМЫ УПРАВЛЕНИЯ ОБУЧАЕМОГО В ВУЗЕ

3.1 Разработка функциональной модели предметной области

Для функционального моделирования адаптивной системы управления обучением в ВУЗе использована диаграмма вариантов использования, являющаяся частью унифицированного языка моделирования (UML).

Общий вид обобщённой модели предметной области обучающей системы представлен на рисунке 3.1.

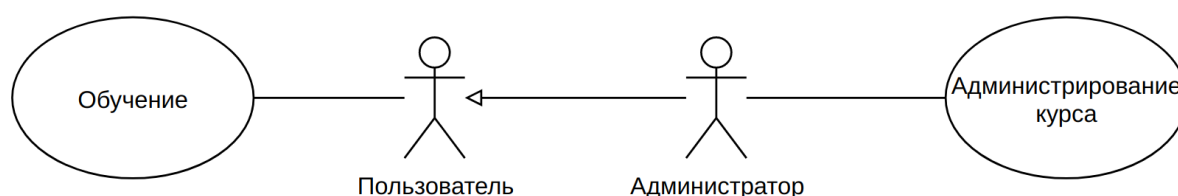


Рисунок 3.1 – Обобщённая модель предметной области

Как видно из модели на рисунке 3.1, используется два актёра:

1. пользователь — это любой конечный пользователь из круга лиц, имеющий доступ к адаптивной обучающей системе;
2. администратор — это такой пользователь, который кроме всех обычных функций, доступных пользователю, имеет ещё специфические функции создания и настройки курсов обучения; следует отметить, что администратор унаследован от пользователя.

Главной функцией пользователя является обучение. У администратора следует выделить функцию, которая позволяет создать, и настраивать курс обучения.

Модель детализации функции «Обучение» представлена на рисунке 3.2.

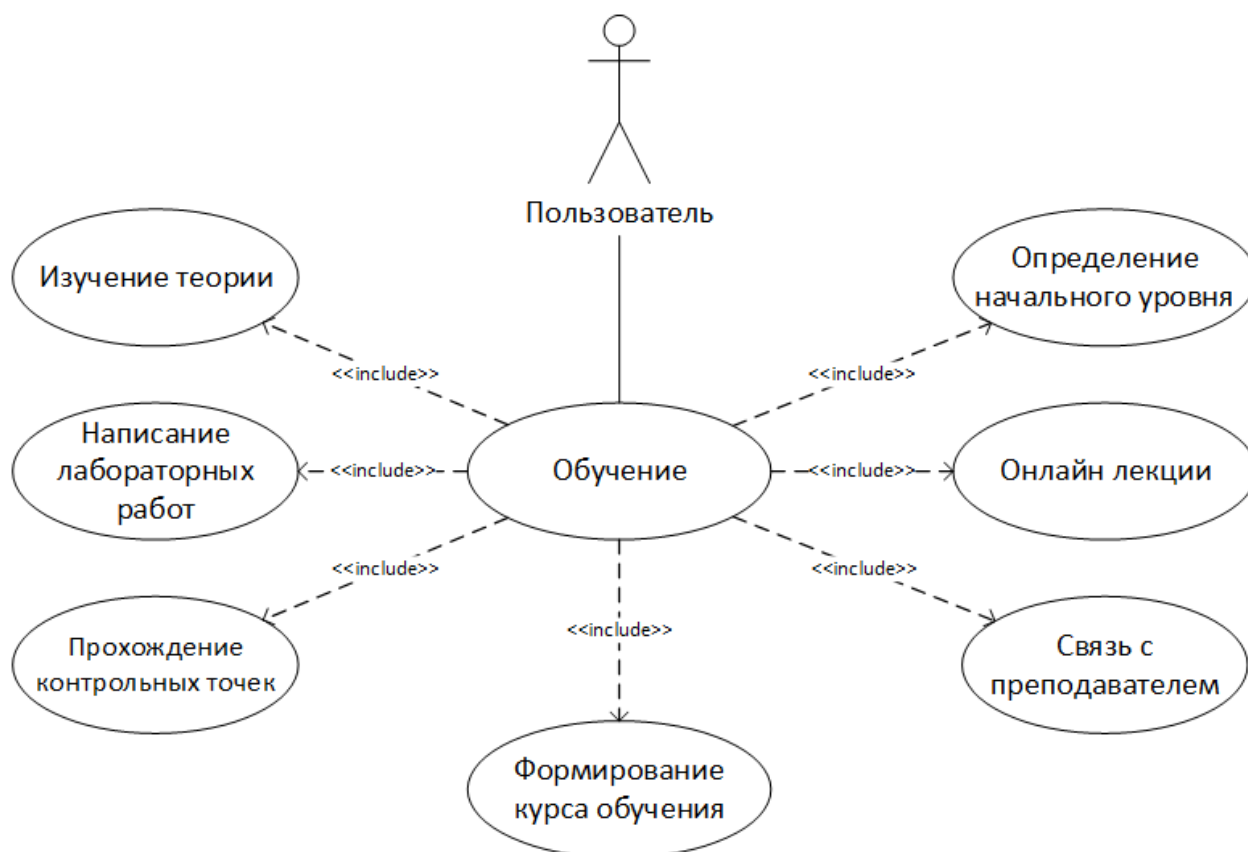


Рисунок 3.2 – Модель детализации функции «Обучение»

Как видно из модели на рисунке 3.2, функция «Обучение» включает в себя следующие функции:

1. Формирование курса обучения: функция формирования курса обучения означает возможность пользователем сформировать курс обучения, основываясь на свой начальный уровень, а также на свои предпочтения;
2. Определение начального уровня: функция определения начального уровня возможность пользователем пройти тест, чтобы узнать уровень обучающегося и в дальнейшем скорректировать программу курса, пользуясь сетевым графиком учебного плана;
3. Изучение теории: функция изучение теории означает возможность пользователем получить доступ ко всей теоретической информации по выбранному курсу. Данные по курсу хранятся в иерархической форме и автоматически адаптируются для предпочтений пользователя;
4. Онлайн-лекции: функция онлайн-лекции означает возможность пользователем быть участником онлайн-лекции с использованием стриминговых платформ;
5. Связь с преподавателем: функция связь с преподавателем означает возможность пользователем поддерживать непосредственный контакт с преподавателями посредством переписки, или онлайн звонков;
6. Написание лабораторных работ: функция написания лабораторных работ означает возможность пользователем выполнять лабораторные

работы и мгновенно получать обратную связь от программы;

7. Прохождение контрольной точки: функция прохождения контрольной точки означает возможность пользователю пройти тест, по результатам которого будут сформированы результаты, на основании которых можно будет решить пройден ли курс пользователем.

Модель детализации функции «Администрирование курса», доступной администратору, представлена на рисунке 3.3.

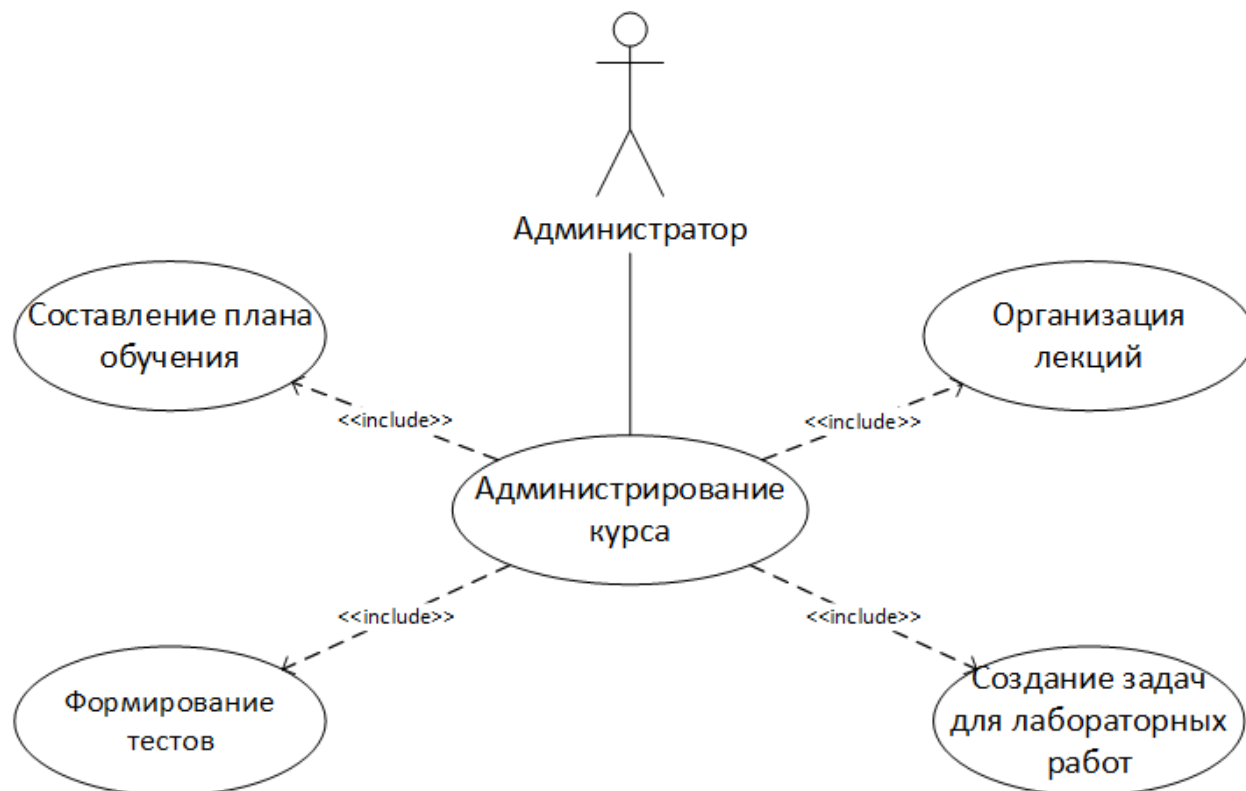


Рисунок 3.3 – Модель детализации функции «Администрирование курса»

Как видно из модели на рисунке 3.3, функция «Администрирование курса» включает в себя следующие функции:

1. Составление плана обучения: функция составления плана обучения предназначена для определения администратором структуры курса, определения временных рамок, формирование иерархии зависимых курсов;

2. Формирование тестов: данная функция позволяет администратору создать тесты, которые в дальнейшем обучаемый будет проходить. В тесте могут быть как открытые, так и закрытые вопросы;

3. Создание задач для лабораторных работ: функция заключается в возможности администратором создавать задания для лабораторных работ, выставлять требования к ним;

4. Организация лекций: данная функция позволит администратору запланировать лекцию и автоматически разослать уведомление всем пользователям, которые подписаны на данный курс.

3.2 Разработка укрупнённой архитектуры программного средства

Как было отмечено выше, разрабатываемая адаптивная система подразумевает отсутствие установленных специфических программных средств на компьютере пользователя. Все запросы удалённо через локальную сеть или сеть Интернет поступают с компьютера пользователя на сервера, где выполняются соответствующие операции. Соответственно, наиболее подходящей архитектурой для программного средства является архитектура клиент-сервер, полностью отделяющая клиентскую часть от серверной. Ниже представлены достоинства подобной архитектуры с точки зрения разработки программных средств:

- отсутствие дублирования кода программы-сервера программами-клиентами;
- снижение требований к компьютерам, на которых установлена клиентская часть, так как все вычисления выполняются на сервере;
- высокая защищённость сервера, на котором хранятся все данные, на сервере проще обеспечить контроль полномочий, чтобы разрешать доступ к данным только клиентам с соответствующими правами доступа;
- возможность объединить различные клиенты: использовать ресурсы одного сервера часто могут клиенты с разными аппаратными платформами, операционными системами;
- снижение трафика сети за счёт того, что между сервером и клиентом передаются небольшие порции данных.

Недостатки подобной архитектуры можно выделить следующие:

- неработоспособность сервера может сделать неработоспособной всю вычислительную сеть; неработоспособным сервером следует считать сервер, производительности которого не хватает на обслуживание всех клиентов, а также сервер, находящийся на ремонте, профилактике;
- поддержка работы данной системы требует отдельного специалиста — системного администратора;
- довольно высокая стоимость оборудования.

Модель архитектуры программного средства представлена на рисунке 3.4.

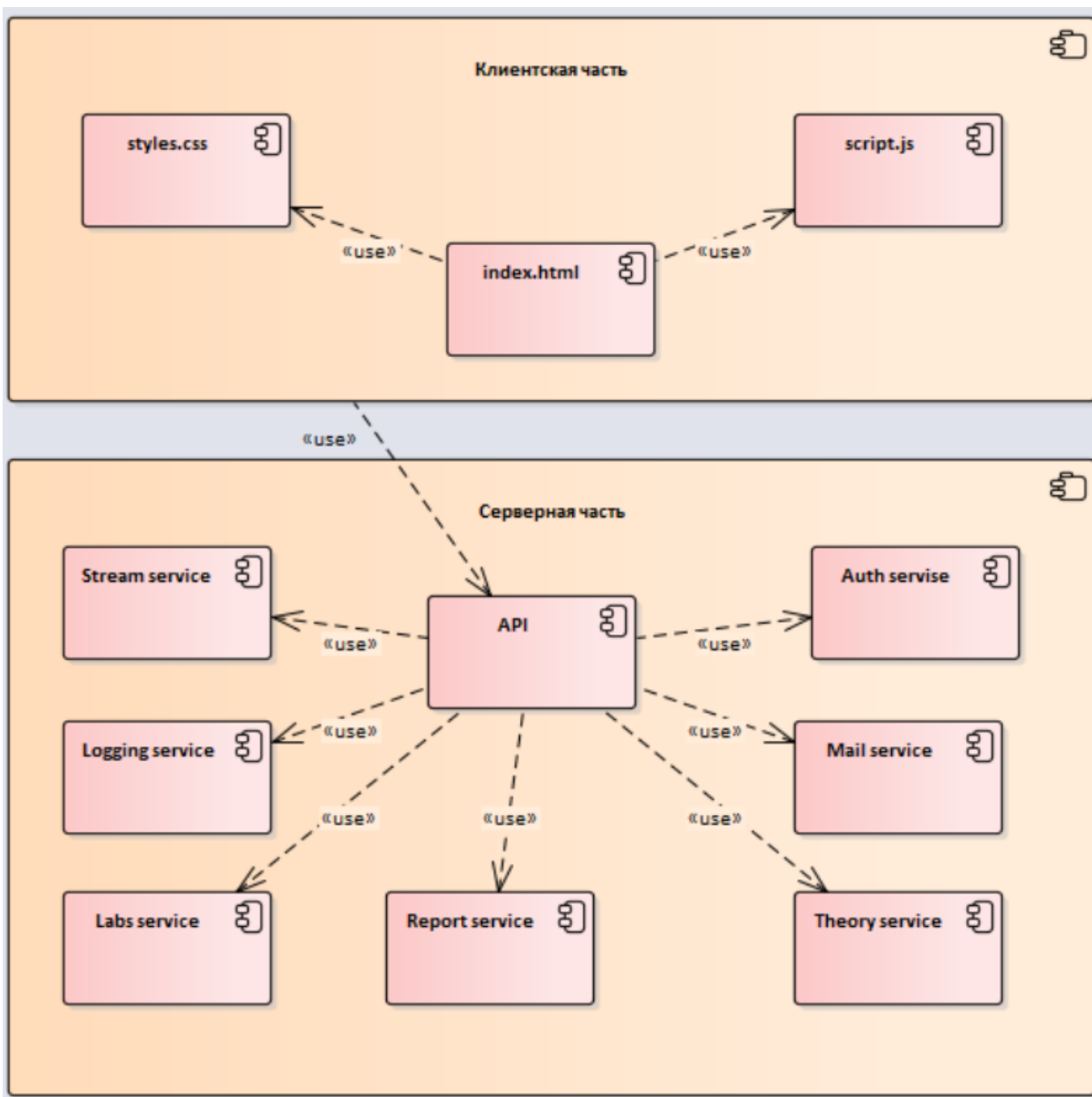


Рисунок 3.4 – Модель архитектуры разрабатываемого программного средства

Как видно из модели архитектуры, клиентская часть представляет собой веб-приложение, состоящее из страниц HTML, таблицей стилей CSS и сценариев на языке JavaScript.

Однако наибольший интерес представляет собой серверная часть, в которой реализованы основные алгоритмы организации учебного процесса. В своей работе клиентская часть отправляет запросы к серверной части. В качестве сервиса на серверной стороне используется сервис API, который содержит принимает все запросы, отправленные клиентом и обрабатывает их.

3.3 Разработка базы данных программного средства

База данных является важнейшим элементом разрабатываемой системы, поэтому качество ее организации повлияет на производительность и безопасность всей системы. В качестве системы управления базой данных была выбрана MongoDB.

MongoDB (от англ. humongous – огромный) – документоориентированная система управления базами данных (СУБД) с открытым исходным кодом, не требующая описания схемы таблиц. Классифицирована как NoSQL, использует JSON-подобные документы и схему базы данных. Написана на языке C++. Данной СУБД присущи следующие основные возможности[45]:

1. MongoDB может работать с набором реплик. Набор реплик состоит из двух и более копий данных. Каждый экземпляр набора реплик может в любой момент выступать в роли основной или вспомогательной реплики. Все операции записи и чтения по умолчанию осуществляются с основной репликой. Вспомогательные реплики поддерживают в актуальном состоянии копии данных. В случае, когда основная реплика дает сбой, набор реплик проводит выбор, который из реплик должен стать основным;

2. MongoDB масштабируется горизонтально, используя шардинг. Пользователь выбирает ключ шарда, который определяет, как данные в коллекции будут распределены. Данные разделятся на диапазоны (в зависимости от ключа шарда) и распределятся по шардам.

3. Может работать в соответствии с парадигмой MapReduce. В фреймворке для агрегации есть аналог SQL-инструкции GROUP BY. Операторы агрегации могут быть связаны в конвейер подобно UNIX-конвейрам. Фреймворк так же имеет оператор \$lookup для связки документов при выгрузке и статистические операции, такие как среднее квадратическое отклонение.

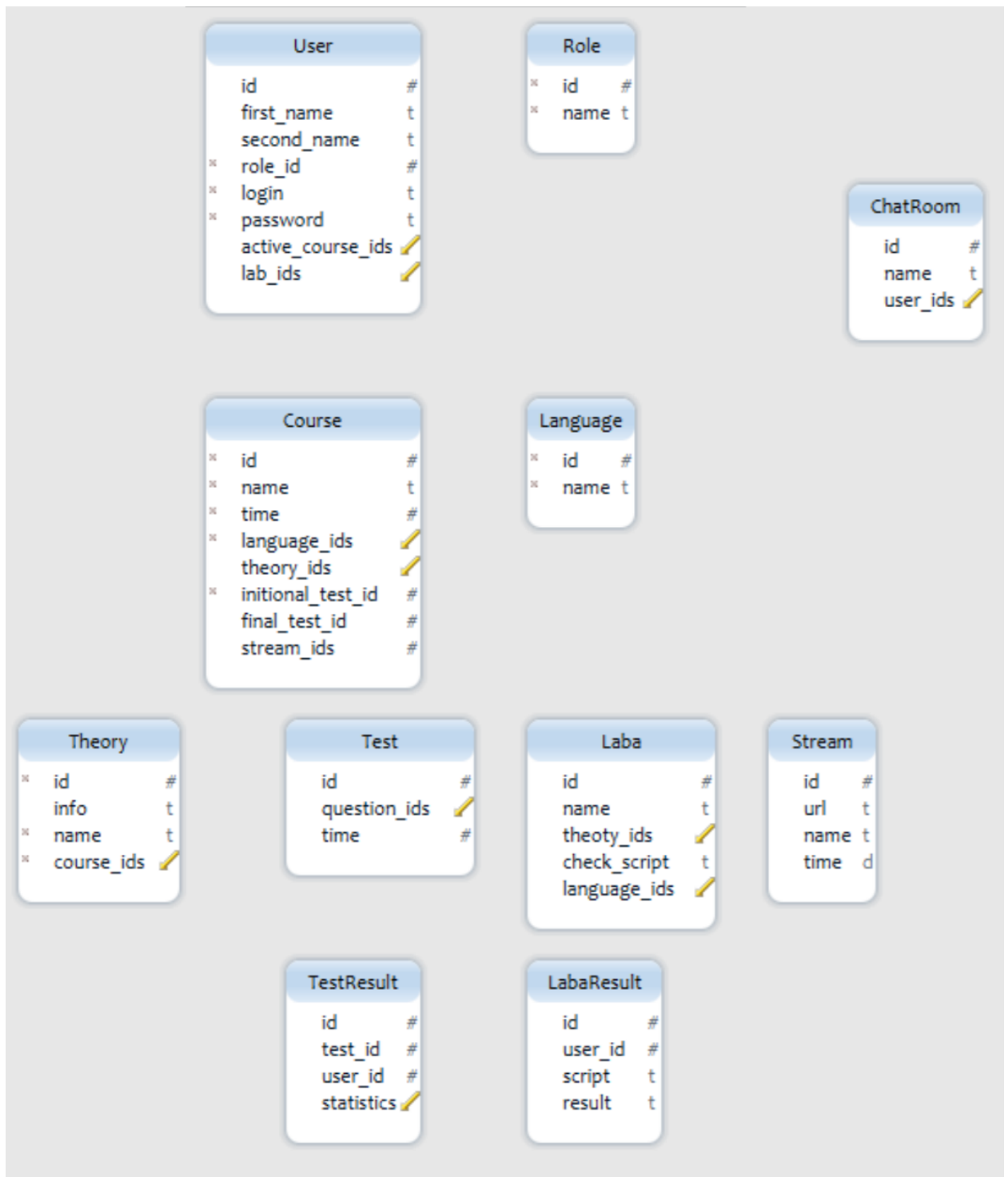


Рисунок 3.5 – Структура базы данных

Для проектирования базы данных используется документный подход. Документная база данных – это тип нереляционных баз данных, предназначенный для хранения частично структурированных данных в виде документов. Документные базы данных интуитивно понятны разработчикам, поскольку данные на уровне приложения обычно представляются как документ JSON. Разработчики могут сохранять данные с помощью той же документной модели, которую они используют в коде приложения. В

документной базе данных все документы могут иметь одинаковую или различную структуру данных. Каждый документ является самоописываемым (т. е. содержит схему, которая может быть уникальной) и не обязательно зависит от какого-либо другого документа. Документы группируются в «коллекции», которые по своему назначению схожи с таблицами в реляционных базах данных.

Для реализации быстрого и удобного поискового движка для теоретической информации и лабораторных работ был выбран Elasticsearch. Elasticsearch – это масштабируемый полнотекстовый поисковый и аналитический движок с открытым исходным кодом. Он позволяет хранить большие объемы данных, проводить среди них быстрый поиск и аналитику в режиме реального времени. Как правило, он используется в качестве базового механизма/технологии, которая обеспечивает работу приложений со сложными функциями и требованиями к поиску. Elasticsearch использует в качестве поисковой основы библиотеку Lucene, которая написана на языке Java и доступна для многих платформ. Все неструктурированные данные хранятся в формате JSON, что автоматически делает ES базой данных NoSQL. Но в отличие от других баз данных NoSQL, ES предоставляет возможности поиска и многие другие функции[46].

3.4 Выводы

1. Разработана функциональная модель предметной области адаптивной системы обучения. Показано, что в данной предметной области используются 2 актёра: пользователь и администратор. У пользователя все операции связаны с обучением, когда у администратора есть функции, позволяющие настраивать процесс обучения;

2. Разработана укрупнённая архитектура программного средства на основе разработанных во второй главе моделей и алгоритмов. Показано, что в модели архитектуры можно выделить такие составные части, как сервер API, кластер микросервисов, и клиентскую часть;

3. Разработана база данных программного средства. Она включает в себя MongoDB базу данных, в которой хранится основная информация системы, и Elasticsearch, служащий для оптимизации поисковых запросов.

ЗАКЛЮЧЕНИЕ

Активное внедрение информационных технологий во все сферы человеческой деятельности предопределяет необходимость использования современных форм подготовки квалифицированных кадров. В свою очередь развитие веб-технологий в целом сделало обучение более доступным. Человек может получить доступ к интересующей его информации моментально. Однако, кроме доступности, развитие технологий позволило улучшить и сам процесс обучения. Одним из таких улучшений является внедрение адаптивности в обучающие системы.

В работе была исследована структура адаптивной обучающей системы, ее компоненты, а также функциональная модель адаптивной обучающей системы. Также были рассмотрены существующие подходы и приложения, на основании которых был произведен сравнительный анализ уже созданных систем адаптивного обучения. Выделены как положительные, так и отрицательные стороны каждой системы. Таким образом, при решении данной задачи была получена необходимая теоретическая база, приняты решения об используемых средствах и подходах при разработке системы и дано обоснование необходимости разработки такой системы.

Были определены основные направления проектирования данной системы. Была разработана модель адаптивной системы, в которую входят следующие элементы: математическая модель пользователя в адаптивной обучающей системе, алгоритм составления учебного плана, а также механизмы адаптации. В качестве результата была получена модель системы, пользуясь которой можно приступить к разработке системы.

Разработана функциональная модель предметной области, описана архитектура системы, выделены основные ее компоненты. Спроектирована база данных, которая позволит масштабировать систему. Приняты решения об использовании определенных технологии и средств для реализации обучающей системы.

Полученные результаты формируют теоретическую и практическую базу для разработки адаптивных систем обучения. Они могут быть использованы для модернизации и дальнейшего развития существующих систем.

В дальнейшем планируется произвести интеграцию описания учебных дисциплин системы с описанием учебных дисциплин из Метасистемы IMS, которая базируется на технологии OSTIS.

Технология OSTIS – одна из технологий искусственного интеллекта для представления знаний в формализованном виде с целью структуризации информации некоторой предметной области. В основе технологии лежит применение в качестве способа представления знаний унифицированных семантических сетей с базовой теоретико-множественной интерпретацией их элементов. Технология OSTIS предоставляет:

– возможность семантической структуризации знаний; учебный материал, изучаемый с помощью интеллектуальной системы учебного назначения, представляется в формализованном виде в качестве раздела базы знаний этой системы. Логико-семантическая структуризация формализованного учебного материала представляется в виде иерархической системы предметных областей соответствующих им онтологий;

– в базу знаний интеллектуальных систем учебного назначения входит целый ряд других предметных областей и онтологий.

Данная технология используется для хранения информации об учебных дисциплинах в виде системы предметных областей по специальности «искусственный интеллект».

Благодаря этому информация об учебных дисциплинах будет представлена в формальном виде.

Также в дальнейшем необходимо добавить поддержку Tin Can API для оптимизации в работе с мобильными устройствами, возможной геймификации процесса обучения. Tin Can API позволяет отслеживать активность и продвижение пользователя даже при отсутствии постоянного интернет соединения, сохраняя информацию на устройстве до возобновления связи с сетью. Кроме того, необходимо учесть, что мы не привязаны к виртуальному обучению и с помощью Tin Can API у нас появится возможность отслеживать любые события, которые кажутся нам частью процесса обучения. Tin Can помогает совместить цифровое обучение с обучением в реальном мире с помощью самостоятельного занесения информации учителями и учениками.

Все это поможет улучшить адаптационные процессы и создать интеллектуальную адаптивную обучающую систему.

Таким образом, цель магистерской диссертации достигнута, задачи выполнены в полном объёме.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Филлипов, Е. Н. Программная инженерия и педагогические программные средства / Е. Н. Филлипов // Системы и средства информатики. — 1992. — № 8.
- [2] Fitzgerald, M. Toward a model of distributed learning / M. Fitzgerald. — Философские проблемы образования, 2002.
- [3] Растринин, Л. А. Адаптивное обучение с моделью обучаемого / Л. А. Растринин, М. Х. Эренштейн. — Рига : Зинатне, 1988.
- [4] Beaumont, Ian H. User modelling in the interactive anatomy tutoring system ANATOM-TUTOR / Ian H Beaumont // User Modeling and User-Adapted Interaction. — 1994. — Vol. 4, no. 1. — P. 21–45.
- [5] Brusilovsky, Peter. ISIS-Tutor: An intelligent learning environment for CDS/ISIS users / Peter Brusilovsky, L. Pesin // Proceedings of the interdisciplinary workshop on complex learning in computer environments (CLCE94). — Joensuu, Finland: EIC, May 16-19, 1994. — P. 29–33.
- [6] Brusilovsky, Peter. Web-based education for all: a tool for development adaptive courseware / Peter Brusilovsky, John Eklund, Elmar Schwarz // Computer networks and ISDN systems. — 1998. — Vol. 30, no. 1-7. — P. 291–300.
- [7] Brusilovsky, Peter. Methods and techniques of adaptive hypermedia / Peter Brusilovsky // Adaptive hypertext and hypermedia. — Springer Netherlands, 1998. — P. 1–43.
- [8] De Bra, Paul. АНА! Adaptive Hypermedia for All / Paul De Bra, Jan-Peter Ruiters // Proceedings of the AACE WebNet Conference. — 2001. — P. 262–268.
- [9] Fischer, Gerhard. User modeling in human-computer interaction / Gerhard Fischer // User modeling and user-adapted interaction. — 2001. — Vol. 11, no. 1-2. — P. 65–86.
- [10] Kobsa, Alfred. Generic user modeling systems / Alfred Kobsa // User modeling and user-adapted interaction. — 2001. — Vol. 11, no. 1-2. — P. 49–63.
- [11] Mental Models and User Models / R. B. Allen, M. G. Helander, T. K. Landauer, P. V. Prabhu // Handbook of Human-Computer Interaction. — 1997. — Vol. 1. — P. 49–63.

- [12] Cobern, William W. Contextual constructivism: The impact of culture on the learning and teaching of science / William W Cobern // *The Practice of Constructivism in Science Education*. — Routledge, 2012.
- [13] Perrault, C Raymond. A plan-based analysis of indirect speech acts / C Raymond Perrault, James F Allen // *Computational Linguistics*. — 1980. — Vol. 6, no. 3-4. — P. 167–182.
- [14] Rich, Elaine. Stereotypes and User Modeling / Elaine Rich // *User Models in Dialog Systems*. — Springer Berlin Heidelberg, 1989. — P. 35–51.
- [15] Кольцов, Ю. В. Нейросетевые модели в адаптивном компьютерном обучении / Ю. В. Кольцов, Н. Ю. Добровольская // *Образовательные технологии и общество*. — 2002. — Т. 5, № 2. — с. 213–216.
- [16] An ontology-based intelligent authoring tool / Weiqin Chen [et al.] // *Proceedings of ICSE'98*. — 1998. — P. 41–49.
- [17] Теоретические основы разработки модели специалиста: В помощь слушателям факультета новых методов и средств обучения при Политехническом музее / Н. Ф. Талызина [и др.]. — Знание, 1986.
- [18] Талызина, Н. Ф. Управление процессом усвоения знаний. Изд. 2-е / Н. Ф. Талызина // Москва. — 1984. — 30 с.
- [19] Шабалина, О. А. Модели и методы для управления процессом обучения с помощью адаптивных обучающих систем / О. А. Шабалина // *Открытое образование и информационные технологии: Материалы Всероссийской научно-методической конференции (приложение к журналу "Открытое образование")*. — 2005. — oct. — с. 306–310.
- [20] Weber, Gerhard. Adaptive learning systems in the World Wide Web / Gerhard Weber // In J. Kay (Ed.), *User modeling: Proceedings of the Seventh International Conference, UM99* / Citeseer. — Vienna: Springer, 1999.
- [21] Hockemeyer, Cord. RATH – A relational adaptive tutoring hypertext WWW-environment based on knowledge space theory / Cord Hockemeyer, Theo Held, Dietrich Albert. — 1998. — Vol. 98. — P. 417–423.
- [22] Chua, Bee Bee. Applying the ISO 9126 model to the evaluation of an e-learning system / Bee Bee Chua, Laurel Evelyn Dyson // *Proc. of ASCILITE*. — 2004. — P. 5–8.
- [23] Schwarz, Elmar. World-Wide Intelligent Textbooks / Elmar Schwarz, Peter Brusilovsky, Gerhard Weber // *i-manager's Journal of Educational Technology*. — 2005. — mar. — Vol. 1, no. 4. — P. 23–28.

[24] Herder, Eelco. Utility-based evaluation of adaptive systems / Eelco Herder // The proceedings of the Second Workshop on Empirical Evaluation of Adaptive Systems, at the 9th International Conference on User Modeling, UM2003, Pittsburg, USA. — 2003. — P. 25–30.

[25] de Rosis, Fiorella. User tailored hypermedia explanations / Fiorella de Rosis, Berardina De Carolis, Sebastiano Pizzutilo // INTERCHI. — Vol. 93. — 1993. — P. 169–170.

[26] Boyle, Craig. MetaDoc: An Adaptive Hypertext Reading System / Craig Boyle, Antonio O Encarnacion // Adaptive Hypertext and Hypermedia. — Springer Netherlands, 1998. — P. 71–89.

[27] Henze, Nicola. Bayesian modeling for adaptive hypermedia systems / Nicola Henze, Wolfgang Nejdl // ABIS 99, 7. GI-Workshop Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen / Citeseer. — 1999.

[28] Hohl, Hubertus. Hypadapter: An adaptive hypertext system for exploratory learning and programming / Hubertus Hohl, Heinz-Dieter Boecker, Rul Gunzenhaeuser // User Modeling and User-Adapted Interaction. — 1996. — jul. — Vol. 6, no. 2-3. — P. 131–156.

[29] Campbell, Brad. HAM: a general purpose hypertext abstract machine / Brad Campbell, Joseph M Goodman // Communications of the ACM. — 1988. — Vol. 31, no. 7. — P. 856–862.

[30] Furuta, Richard. The Trellis hypertext reference model / Richard Furuta // Proc. Hypertext Standardization Workshop / National Institute of Standards and Technology. — 1990. — P. 83–94.

[31] Halasz, Frank. The Dexter hypertext reference model / Frank Halasz, Mayer Schwartz // Communications of the ACM. — 1994. — feb. — Vol. 37, no. 2. — P. 30–39.

[32] Wu, Hongjing. A reference architecture for Adaptive Hypermedia Applications / Hongjing Wu. — Citeseer, 2002.

[33] De Bra, Paul. AHAM: a Dexter-based reference model for adaptive hypermedia / Paul De Bra, Geert-Jan Houben, Hongjing Wu // Proceedings of the tenth ACM Conference on Hypertext and hypermedia: returning to our diverse roots: returning to our diverse roots. — 1999. — P. 147–156.

[34] Kay, Judy. An individualised course for the C programming language / Judy Kay, RJ Kummerfeld // Proceedings of Second International WWW Conference. — 1994. — P. 17–20.

[35] Baronas, J. Hypertext Standardization Workshop - Gaithersburg, MD - January 16-18, 1990 / J. Baronas // Journal of Research of the National Institute of Standards and Technology. — 1990. — may. — Vol. 95, no. 3. — P. 345–348.

[36] INSPIRE: An INtelligent System for Personalized Instruction in a Remote Environment / Kyparisia A. Papanikolaou, Maria Grigoriadou, Harry Kornilakis, George D. Magoulas // Hypermedia: Openness, Structural Awareness, and Adaptivity. — Springer Berlin Heidelberg, 2002. — P. 215–225.

[37] Finin, Tim. GUMS: A general user modeling system / Tim Finin, David Drager // Proceedings of the workshop on Strategic computing natural language / Association for Computational Linguistics. — 1986. — P. 224–230.

[38] Brusilovsky, Peter. Adaptive educational systems on the world-wide-web: A review of available technologies / Peter Brusilovsky [et al.] // Proceedings of Workshop "WWW-Based Tutoring" at 4th International Conference on Intelligent Tutoring Systems (ITS'98), San Antonio, TX. — 1998.

[39] Henze, Nicola. Modeling constructivist teaching functionality and structure in the KBS Hyperbook System / Nicola Henze, Wolfgang Nejdl, Martin Wolpers // Proceedings of the 1999 conference on Computer support for collaborative learning - CACL '99. — International Society of the Learning Sciences (ISLS), 1999. — jul.

[40] Adaptive hyperbooks for constructivist teaching / Nicola Henze, Kabil Naceur, Wolfgang Nejdl, Martin Wolpers // KI. — 1999. — Vol. 13, no. 4. — P. 26–31.

[41] Ullrich, Carsten. Description of an instructional ontology and its application in web services for education / Carsten Ullrich // Proceedings of Workshop on Applications of Semantic Web Technologies for E-learning, SW-EL. — Vol. 4. — 2004. — P. 17–23.

[42] Шабалина, О. А. Модели и методы для управления процессом обучения с помощью адаптивных обучающих систем: Дис.канд. техн. наук / О. А. Шабалина // Астрахань. — 2005.

[43] Карпенко, А. П. Модельное обеспечение автоматизированных обучающих систем. Обзор / А. П. Карпенко, А. А. Добряков // Машиностроение и компьютерные технологии. — 2011. — no. 7.

[44] Кудрявцев, Е. М. Методы сетевого планирования и управления проектом. Учебное пособие / Е. М. Кудрявцев. — 2007.

[45] Mongo [Электронный ресурс]. — Режим доступа: <https://mongodb.org/>
- Дата доступа: 05.03.2020.

[46] Elasticsearch [Электронный ресурс]. — Режим доступа: <https://elastic.co/> - Дата доступа: 05.03.2020.