

Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«ВОЛГОГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

На правах рукописи



Хайров Александр Валерьевич

**МОДЕЛИ И МЕТОДЫ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ
АДАПТИВНЫХ ОБУЧАЮЩИХ ИГР**

2.3.7. Компьютерное моделирование и автоматизация проектирования

ДИССЕРТАЦИЯ

на соискание ученой степени кандидата технических наук

Научный руководитель:

доктор технических наук, профессор

Садовникова Наталья Петровна

Волгоград – 2025

Оглавление

Введение.....	5
Глава 1. Научно-методические основы проектирования адаптивных обучающих игр	12
1.1 Эволюция и современные тенденции развития обучающего ПО	12
1.2 Модели и методы проектирования адаптивных обучающих систем	15
1.2.1 Модели предметной области.....	15
1.2.2 Модели обучаемого.....	16
1.2.3 Методы сбора информации об обучаемом	17
1.2.4 Методы адаптации	19
1.3 Модели адаптации для адаптивных обучающих игр	23
1.4 Встроенные модели адаптации для адаптивных обучающих игр.....	26
1.5 Инструменты проектирования адаптивного обучающего ПО	33
1.6 Выводы по главе	36
Глава 2. Разработка переносимой модели адаптации для адаптивных обучающих игр	38
2.1 Модель предметной области	40
2.2 Модель обучаемого	47
2.3 Метод адаптации	48
2.4 Выводы по главе	53
Глава 3. Автоматизация проектирования адаптивных обучающих игр	55
3.1 Процесс автоматизированной разработки адаптивных обучающих игр ..	55
3.2 Шаблон архитектуры адаптивных обучающих игр	58
3.3 Модуль адаптации	59
3.4 Разработка платформозависимого набора решений модуля адаптации ...	60

3.5 Анализ игровых движков	61
3.6 Разработка платформозависимого набора решений.....	62
3.6.1 Реализация метода триггеров.....	63
3.6.2 Реализация системы диалоговых окон.....	65
3.6.3 Разработка БД.....	67
3.7 Интеграция модуля адаптации с набором игровых решений.....	69
3.8 Разработка метода взаимодействия проектировщиков обучающего контента и разработчиков адаптивной обучающей игры.....	70
3.9 Метод автоматизированного проектирования адаптивных обучающих игр.....	71
3.10 Выводы по главе.....	72
Глава 4. Апробация моделей и методов на разработке адаптивной обучающей игры	73
4.1 Прототип обучающей игры.....	73
4.2 Ad-hoc разработка адаптивной обучающей игры.....	74
4.2.1 Разработка набора игровых решений.....	75
4.2.2 Разработка нелинейной структуры обучающего курса.....	78
4.2.3 Построение пространства знаний.....	81
4.2.4 Интерпретация обучающих действий в игровом контексте.....	82
4.2.5 Реализация модели адаптации и настройка механизмов адаптации	83
4.3 Разработка адаптивной обучающей игры с применением метода автоматизированного проектирования	86
4.3.1 Описание обучающего контента адаптивной обучающей игры в виде матрицы смежности	88
4.3.2 Описание фрагментов обучающего контента в игровом контексте	88
4.3.3 Построение пространства знаний.....	88

4.3.4 Разработка архитектуры адаптивной обучающей игры.....	90
4.3.5 Интеграция модуля адаптации с набором игровых решений и настройка механизмов адаптации.....	90
4.4 Апробация метода автоматизированного проектирования на других адаптивных обучающих играх	94
4.5 Выводы по главе.....	97
Глава 5. Оценка качества и снижение трудоемкости разработки адаптивных обучающих игр	98
5.1 Методы оценки качества адаптивных обучающих игр.....	98
5.2 Метод эвристических оценок качества для оценки качества обучающего и адаптивного обучающего ПО	101
5.3 Адаптация метода эвристических оценок для оценки качества АОИ.....	104
5.4 Сравнительная оценка качества адаптивной обучающей игры	106
5.5 Оценка временных затрат.....	109
5.6 Выводы по главе.....	112
Заключение	114
Список литературы	116
ПРИЛОЖЕНИЕ А. Пример построения пространства с пояснением фрагментов кода	129
ПРИЛОЖЕНИЕ Б. Копия свидетельства о государственной регистрации программы ЭВМ.....	136
ПРИЛОЖЕНИЕ В. Копия акта о внедрении результатов исследования	137

Введение

Актуальность. К обучающему программному обеспечению (ПО) относят программные продукты, главным назначением которых является обучение и/или развитие некоторых навыков. Обучающие системы являются одним из наиболее динамично развивающихся видов ПО, востребованность в которых постоянно возрастает.

Стремление к повышению качества обучающего ПО привело к появлению адаптивных обучающих систем, т.е. систем, обеспечивающих персонализацию процесса взаимодействия пользователя с системой. Сфера разработки адаптивных обучающих систем активно развивается, хорошо систематизирована, разработаны модели, методы и технологии, применимые для адаптации процесса обучения в таких системах.

Новым шагом в сфере разработки обучающих систем является персонализация процесса обучения в игровом контексте, т.е. разработка адаптивных обучающих игр. Однако прямой перенос моделей, методов и технологий адаптации, разработанных для адаптивных обучающих систем, в адаптивные обучающие игры может существенно ограничивать обучающие возможности и/или снижать привлекательность обучающей игры с точки зрения игровой компоненты. Поэтому в настоящее время превалирует подход, рассматривающий адаптивные обучающие игры как самостоятельную категорию ПО, требующую разработки оригинальных моделей, методов и технологий проектирования, учитывающих специфику этой категории ПО.

Известные модели адаптации, разработанные специально для адаптивных обучающих игр, являются встроенными *ad-hoc* моделями, привязанными к реализации одной игры. Поэтому актуальной является задача разработки переносимой модели адаптации, интегрируемой в проекты по разработке адаптивных обучающих игр различных жанров, что позволит автоматизировать процесс проектирования адаптивных обучающих игр, снизить трудоемкость разработки таких игр и повышать их качество.

Степень изученности проблемы. Исследованиями способов формализации процесса обучения для разработки обучающего ПО занимаются российские и зарубежные ученые. Существенный вклад в развитие этого направления исследований внесли М.В. Лукьяненко, И.С. Казаков, А.Ю. Аксенова, Н.В. Примчук, S. Sfenrianto, А.Н. Zinal, S. Heru, В. Boyn, M. Julia, G. Lourdes, E. Festhi. Различные аспекты моделирования процессов обучения рассматриваются в работах А.И. Вершинина, Б.Т. Солдатов, Ю.О. Герман, О.В. Герман, К. Е. Plamondon, M. A. Donovan, E.D. Pulakos, S. Arad.

Одним из приоритетных направлений исследований является разработка способов моделирования областей знаний, применимых для адаптивных обучающих систем. Исследованиям в этой области посвящены работы Л.С. Narasimha, В. Henderik, P.A. Chen, N. Kinshuk, С. Wei, Н. Chen, В. Martin, А., Mitrovic, K.R. Koedinger, S. Mathan. Модели обучаемого предложены в работах В.А. Чулюкова, Н. Tairi, N. Loc, Т. Barnes, Н. Cen, К. Koedinger, В.Л. Junker, S. Chen, P.-J. Sue, R. Lopes, R. Bidarra; модели предметной области – в работах К.В. Беляева, А.Н. Афанасьева, Ю.В. Вайнштейна, M. Jipp, L. Xianglong, L.C. Tan, G. Lianli, R. Houlette, L. Patricia, Z. Ling; модели адаптации – в работах О. А. Шабалиной, А. М. Hussain, K. Sehaba, А. Т. Mille, О. Tumenayu, С. Reuter, F. Mehm, S. Göbel, R. Steinmetz, S. Göbel, F. Mehm, S. Radke, R. Steinmetz, M. Morizane, Y. Nakano, K. Shimohara, I. E. Tanev, Jr.-E. D. Lavieri.

Исследованиями, связанными с автоматизацией проектирования обучающего и адаптивного обучающего ПО, занимались Т. В. Тулупова, А. А. Тищенко, П. Л. Цытович, Г. Н. Зверев, В. S. Afolabi, Yu.-W. Lin, R. Sottolare, M. K. Idrissi.

Целью диссертационной работы является повышение качества адаптивных обучающих игр и снижение трудоемкости их разработки за счет автоматизации процесса проектирования таких игр и применения готовых проектных и программных решений. Для достижения поставленной цели выделены следующие задачи:

1. Провести анализ существующих моделей, методов и технологий разработки обучающего ПО и их применимости к разработке адаптивных обучающих игр.
2. Разработать переносимую модель адаптации, применимую для построения адаптивных обучающих игр различных жанров.
3. Разработать технологию автоматизированного проектирования адаптивных обучающих игр на основе переносимой модели адаптации.
4. Спроектировать и реализовать встраиваемый модуль адаптации, реализующий разработанную модель адаптации для адаптивных обучающих игр.
5. Провести апробацию модуля адаптации на разработке адаптивных обучающих играх разных жанров.
6. Оценить качество разработанных адаптивных обучающих игр с помощью метода эвристических оценок и трудоемкость разработки игры.

Объектом исследования является процесс проектирования адаптивных обучающих игр.

Предметом исследования являются способы автоматизации проектирования адаптивных обучающих игр.

Методы исследования: в диссертационном исследовании использованы методы алгебраической теории систем, теории множеств, теории графов, теории пространств знаний (*Knowledge Space Theory, KST*), численные методы линейной алгебры, методы автоматизации проектирования.

Научная новизна заключается в разработке нового метода автоматизированного проектирования адаптивных обучающих игр, включающего следующие новые научные результаты:

1. Разработана модель адаптации, включающая модель предметной области в игровом контексте, модель обучаемого/игрока и метод адаптации, отличающаяся от известных моделей адаптации переносимостью и интегрируемостью с играми различных жанров [п.п.3].

2. Впервые формализованы проектные и технологические процедуры процесса разработки адаптивных обучающих игр, представляемые алгоритмами построения игрового сценария, управления процессом обучения в игровом контексте и интеграции модели адаптации с набором игровых решений, позволяющие автоматизировать процесс проектирования адаптивных обучающих игр различных жанров и сократить состав проектных процедур [п.п.1,3].
3. Разработан шаблон архитектуры адаптивных обучающих игр, отличающийся составом платформозависимых и платформонезависимых компонент, интегрируемых с игровыми проектами, позволяющий снизить трудоемкость этапа проектирования адаптивных обучающих игр [п.п.1].
4. Предложен метод взаимодействия «Проектировщик – система», отличающийся возможностью совместной разработки модели адаптации проектировщиками обучающего контента и разработчиками обучающей игры с применением программного модуля адаптации [п.п.4].

Основные положения, выносимые на защиту:

1. Переносимая модель адаптации, применимая для проектирования адаптивных обучающих игр различных жанров.
2. Алгоритмы построения игрового сценария, управления процессом обучения в игровом контексте и интеграции модели адаптации с набором игровых решений, позволяющие автоматизировать процесс проектирования адаптивных обучающих игр;
3. Шаблон архитектуры адаптивных обучающих игр, включающий набор платформозависимых и платформонезависимых компонент, интегрируемых с игровыми решениями.
4. Метод совместной разработки модели адаптации проектировщиками обучающего контента и разработчиками обучающей игры.
5. Встраиваемый модуль адаптации, реализующий предложенную модель адаптации для разработки адаптивных обучающих игр различных жанров.

Степень достоверности включенных в исследование научных положений, теоретических выводов, практических рекомендаций обусловлена корректным применением указанных методов исследования и успешным практическим применением результатов диссертационной работы, что отражено в актах внедрения.

Теоретическая и практическая значимость. Предложенные модели и методы развивают методологические основы разработки адаптивных обучающих игр и могут быть использованы как теоретический базис для разработки новых моделей и методов адаптации, применимых для разработки адаптивного обучающего ПО.

Разработанные проектные, архитектурные и технологические решения применимы для проектирования и реализации адаптивных обучающих игр различных жанров.

Апробация работы и публикации. По теме диссертации опубликовано 11 работ, в том числе 4 публикации в журналах, рекомендованных ВАК, 3 публикации Scopus/WoS. Получено 1 свидетельство на регистрацию программ ЭВМ.

Основные положения и материалы диссертационной работы докладывались на IV Международной научной конференции «Информационные технологии в науке, управлении, социальной сфере и медицине» в 2019 году, на II международной научно-практической конференции «Научно-техническое развитие России и мира» в 2023 году, на European Conference on Games Based Learning, ECGBL в 2023 г., на V Международной научно-практической конференции “Creativity in Intelligent Technologies and Data Science Third Conference, CIT&DS” в 2023 году.

Реализация и внедрение. Проектные и технологические решения, предложенные в диссертационной работе, используются в компании ООО «К-Сервис» для разработки адаптивного программного обеспечения и отдельных программных модулей в составе программных продуктов. Предложенный автором модуль адаптации используется для реализации адаптивных обучающих игр при выполнении курсовых работ, курсовых проектов,

выпускных квалификационных работ бакалавров.

Личный вклад автора. В диссертации представлены результаты исследований, выполненных самим автором или под его непосредственным руководством. Личный вклад автора состоит в постановке задач исследования, разработке проектных и технологических решений, анализе, обобщении полученных результатов и формулировке выводов.

Структура диссертации. Диссертационная работа состоит из введения, пяти глав, заключения, а также библиографического списка из 122 наименований и 3 приложений. Общий объем работы 137 страниц, в том числе 84 рисунка и 18 таблиц.

Во введении проведен анализ подходов к разработке обучающих систем, производится общая характеристика работы: обоснование актуальности проведенных исследований, характеристика научной новизны и практической значимости полученных результатов.

В первой главе изучены тенденции развития адаптивного обучающего ПО, рассмотрены существующие инструментальные средства поддержки разработки адаптивного ПО, определена научная проблема, ее актуальность, цели и задачи исследования.

Во второй главе описана переносимая модель адаптации, разработанная для адаптивных обучающих игр, и ее компоненты: модель предметной области в игровом контексте, модель обучаемого, метод адаптации и контекстная модель.

В третьей главе описаны способы формализации и типизации проектных и технологических процедур и их реализация в технологии автоматизированной разработки адаптивных обучающих игр, и программный модуль, реализующий предложенную модель адаптации.

В четвертой главе описана апробация предложенных моделей, методов и технологий на разработке трех адаптивных обучающих игр: двух игр на движке *Unity* и одной игры на движке *Godot*.

В пятой главе произведена оценка качества разработанных адаптивных обучающих игр и временных затрат на их разработку.

В заключении сформулированы основные научные и практические результаты диссертационной работы.

В приложениях приведен пример построения пространства с пояснением фрагментов кода, копия свидетельства о государственной регистрации и программы для ЭВМ и копия акта о внедрении результатов исследования.

Автор выражает благодарность Шабалиной Ольге Аркадьевне, кандидату технических наук, доценту кафедры «САПР и ПК» ВолгГТУ, за оказанную помощь в разработке модели адаптации для адаптивных обучающих игр и консультации в ходе выполнения диссертационной работы.

Глава 1. Научно-методические основы проектирования адаптивных обучающих игр

1.1 Эволюция и современные тенденции развития обучающего ПО

Обучающее (образовательное) программное обеспечение (обучающее ПО, educational software) включает все виды ПО, созданного для образовательных целей [1,2]. Исследования показывают [3,4], что применение обучающего ПО для освоения знаний и тренировки различных навыков положительно влияет на когнитивное, поведенческое и эмоциональное развитие обучаемых.

Сфера разработки обучающего ПО давно и активно развивается и хорошо систематизирована [5]. К настоящему времени накоплен большой опыт, как в области теоретических исследований, так и в области практической разработки обучающих систем [6-11].

Ключевым этапом разработки обучающей системы является разработка способа представления изучаемой области знаний. Для представления знаний в обучающих системах используют модели предметной области (Knowledge Domain model) [12]. В большинстве случаев, модель предметной области представляется графом общего вида, задающим конечное множество фрагментов знаний, составляющих содержание области знаний, и связи между фрагментами, отражающие логику ее освоения. Для отражения структурных особенностей области знаний граф может наделяться дополнительными. Некоторые обучающие системы сохраняют также информацию о динамике состоянии пользователя в процессе взаимодействия с системой в модели (профиле) пользователя (Learner model, Learner profile). Но стратегия обучения в обучающих системах задаётся априори и не зависит от состояния пользователя и текущих результатов его обучения.

Стремление к повышению эффективности обучающего ПО привело к появлению адаптивных обучающих систем (adaptive learning systems, ALS), обеспечивающих персонализацию процесса обучения [13]. Адаптивные обучающие системы динамически изменяют содержание, методы и темп обучения

в зависимости от текущих результатов и характеристик обучаемых и обеспечивают таким образом персонализацию процесса обучения для каждого пользователя системы [14]. Адаптивные обучающие системы позволяют учитывать различные стили обучения, уровень подготовки и интересы обучаемых [15] и помогают обучаемым ориентироваться в большом количестве информации и эффективно осваивать новые навыки [16].

Современные подходы к разработке адаптивных систем основаны на выделении трех составляющих: модель предметной области, модель обучаемого и адаптационная модель (Adaptation model). Адаптационные модели относятся к классу поведенческих моделей. Известные подходы к построению таких моделей в большинстве своем основаны на правилах, причинно-следственных связях, т.е. способах, основанных на механизмах логического вывода. Определение логики адаптации зависит от выбранной модели обучаемого.

Параллельно с развитием обучающих систем активно развивалась и игровая индустрия. Компьютерные игры представляют собой также хорошо изученный самостоятельный класс ПО. Данные в компьютерной игре классифицируются по следующим группам: механики игры, контент игры, данные профиля игрока, настройки игры, общая игровая статистика, состояние игры (прогресс). Способы формализации и хранения данных зависят от конкретной реализации игры. Контент игры – это заранее определенные ресурсы: графические и аудиофайлы, сценарий. Профиль игрока содержит данные, которые вводит сам пользователь и которые практически не подвергаются изменениям. Настройки задают внешние параметры игрового процесса, такие как сложность или способы отображения информации. Игровая статистика хранит информацию о достижениях игрока за все время, эта информация может быть использована для показа адаптивных подсказок. Изменения в игровом мире и динамические свойства игрока сохраняются в состоянии игры. Если проводить аналогию перечисленных групп с моделями, применяемыми для разработки обучающих систем, то игровой контент по своему назначению соответствует модели предметной области; профиль игрока

и игровая статистика, хранящие данные об игроке, соответствуют модели обучаемого; механика игры может быть соотнесена с моделью процесса обучения.

Компьютерная обучающая игра представляет собой игру, имеющую обучающие цели [17-19]. С другой стороны, обучающая игра может рассматриваться как обучающая система, в которой процесс обучения интегрирован в игру. Таким образом, обучающие компьютерные игры представляют собой объединение двух широко известных классов программных приложений: обучающих систем и компьютерных игр. Соответственно, для организации процесса обучения в обучающих компьютерных играх используют модели предметной области и модели сценария.

Традиционные компьютерные игры для повышения интереса игроков и разнообразности прохождения используют нелинейные сценарии, в которых сюжетная линия не predetermined заранее, а адаптируется к конкретному игроку. Такие игры (адаптивные компьютерные игры), в дополнение к компонентам обычных игр, имеют встроенные механизмы изменения поведения игры в зависимости от текущих игровых результатов пользователя [20]. Изменениям могут подвергнуться параметры игры, сценарий прохождения, контент игры. Простейшим свойством адаптации является изменение параметров: сложности игры или величины награды. С увеличением степени адаптивности может меняться сценарий прохождения игры пользователем. Адаптация контента игры подразумевает качественное изменение отдельных шагов сценария в соответствии с действиями пользователя. Технологии адаптации, применяемые в адаптивных играх, зависят от конкретной реализации игры.

Желание повысить привлекательность обучающих игр и приблизить их по качеству к традиционным играм привело к появлению нового класса обучающих приложений - адаптивным обучающим играм [21]. Адаптивная обучающая игра представляет собой адаптивную обучающую систему, персонализирующую процесс обучения в игре за счет использования нелинейного сюжета [22]. Однако, в отличие от традиционных компьютерных игр, где последовательность заданий ограничивается только логикой повествования, в адаптивных обучающих играх

необходимо учитывать также взаимосвязь обучающих заданий и требуемых для их выполнения знаний. Для разработки адаптивных обучающих игр необходимо согласование моделей адаптации процесса обучения и игрового процесса, охраняющих баланс обучающей и игровой компонент.

Эволюция обучающего ПО представлена на рисунке 1.1 [23].

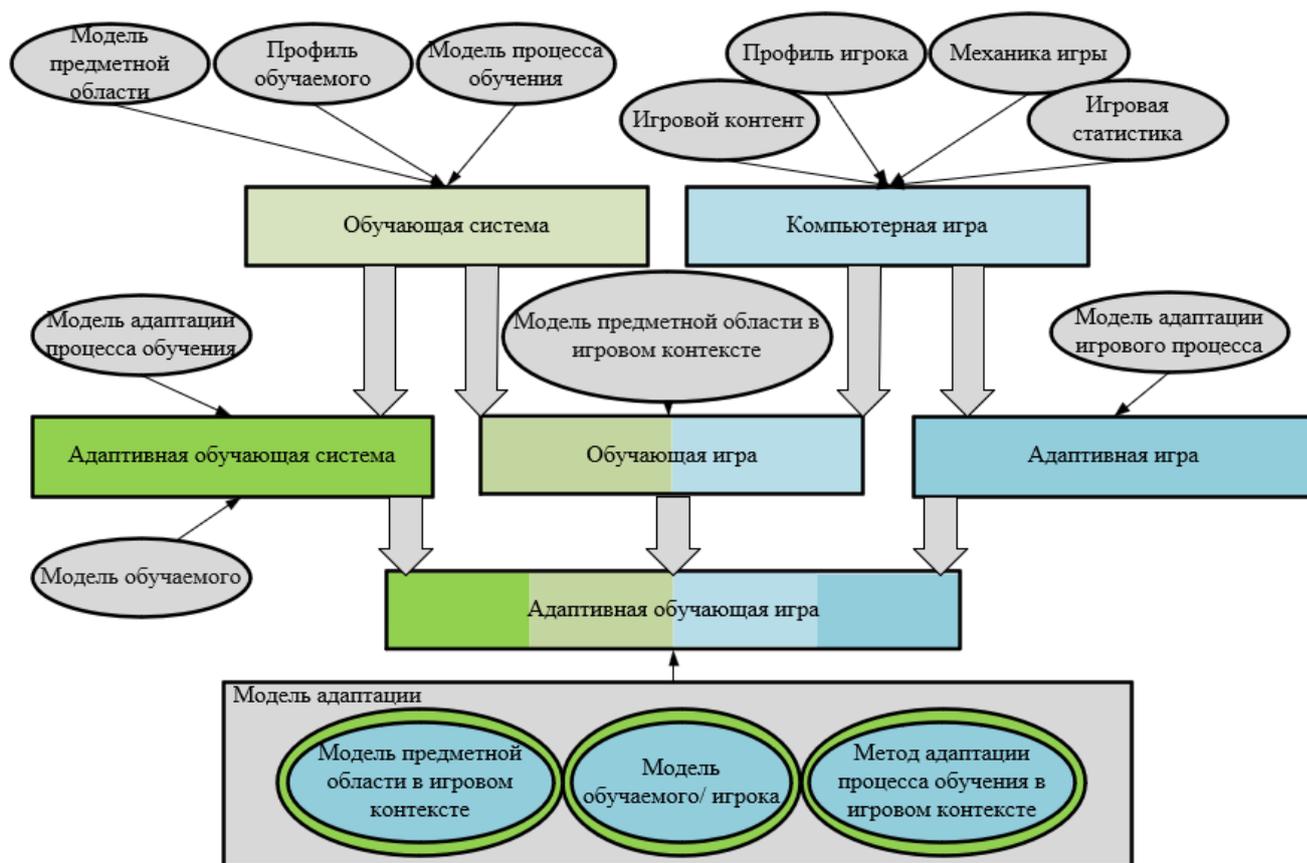


Рисунок 1.1 – Тенденции развития обучающего ПО

1.2 Модели и методы проектирования адаптивных обучающих систем

1.2.1 Модели предметной области

Модели предметной области определяют структуру и содержание материалов, предназначенных для изучения в обучающей системе. Ключевыми аспектами модели предметной области являются [24]:

- структура знаний определяет, какие знания и навыки должны быть представлены в обучающей системе, а также их иерархию и взаимосвязи;

- контент включает в себя текстовые, графические, аудиовизуальные и
- другие формы представления информации, которые используются для обучения;
- адаптивные механизмы позволяют системе изменять представление знаний в зависимости от уровня подготовки и потребностей обучаемого, что делает процесс обучения более персонализированным.

В [25] рассматриваются основные подходы к моделированию предметных областей, используемые в адаптивных обучающих системах. Рассмотрены ключевые парадигмы моделей на основе правил, моделей на основе ограничений, байесовских сетей, моделей машинного обучения, моделей на основе текста, обобщенных примеров и пространств знаний. Показаны взаимосвязи между моделью предметной области и моделью обучаемого, которые являются центральными компонентами многих систем адаптивных обучающих систем.

1.2.2 Модели обучаемого

Модели обучаемого предназначены для хранения характеристик обучаемого, определенных как влияющие на процесс обучения [26,27]. Модель обучаемого может содержать такую информацию о пользователе, как знание предметной области, эффективность обучения, интересы, предпочтения, цель, задачи, опыт, личные качества [28]. Ключевые аспекты модели обучаемого включают [29]:

- уровень знаний: определяет текущее состояние знаний обучаемого в конкретной области, что позволяет системе адаптировать сложность учебного материала;
- стиль обучения: учитывает предпочтения учащегося в отношении способов восприятия информации, таких как визуальный, аудиальный или кинестетический стиль;
- предыдущий опыт: включает информацию о том, какие знания и навыки обучаемый уже освоил, что помогает избежать повторения и сосредоточиться на новых темах;

- мотивация и интересы: учитывает личные интересы обучаемого и его мотивацию к обучению, что может значительно повлиять на вовлеченность и успешность в учебном процессе.

В [30] предложена модель обучаемого, основанная на объединении поведения обучения студентов со степенью понимания концептов знаний изучаемой предметной области. В [31] предложена теоретико-множественная модель обучаемого, включающая пару подмножеств, элементы которых содержат неизменяемую информацию и информацию, переопределяемую на каждой итерации процесса обучения.

Модель обучаемого может содержать информацию, относящуюся к предметной области, и информацию, независимую от предметной области. Информация, относящаяся к предметной области, отражает статус, степень знаний и навыков, которые приобрел обучаемый. Информация, относящаяся к предметной области, организована в виде модели предметной области. В модели обучаемого может храниться дополнительная информация, такая как предварительные знания учащегося, записи об эффективности обучения, записи результатов тестов.

1.2.3 Методы сбора информации об обучаемом

Для адаптации процесса обучения к обучаемому необходимо предоставить системе информацию об обучаемом т.е. определить характеристики обучаемого, которые будут являться параметрами модели. Также необходимо определить, какие характеристики будут считаться статическими, и определяться один раз для каждого пользователя при построении начального курса и какие будут динамическими, т.е. изменяющимися в процессе обучения.

Формирование набора характеристик обучаемого является не тривиальной задачей. Выделяют следующие характеристики об обучаемом [32]:

- уровень знаний;
- психологические характеристики;
- скорость/стиль обучения;

- процент выполнения заданий;
- способность к обучению;
- уровень умений и навыков;
- выбранные метод/стратегия обучения;
- уровень понимания структуры курса.

Методы сбора информации о пользователе разделяют на два типа [33,34]:

- явные методы, направленные на прямое извлечение информации об обучаемом;
- неявные и ненавязчивые методы, позволяющие извлекать информацию о учащих скрытно и ненавязчиво, не нарушая процесс вовлечения в игру.

Основным источником явной информации об обучаемом являются результаты тестового контроля знаний. Поскольку в современных адаптивных обучающих системах используют динамическую модель обучаемого, в процессе тестирования модель обучаемого уточняют и корректируют с учетом фиксируемых показателей его процесса обучения, например, числа и типа ошибок, времени выполнения заданий, отказов от решения, потребности в самоконтроле и помощи и т.д. Существует несколько явных методов сбора информации, которые могут быть использованы в адаптивных обучающих системах [35]:

- опросы и анкеты: этот метод позволяет собрать информацию о предпочтениях, мотивации, стиле обучения и предыдущем опыте обучаемого. Опросы могут проводиться как в начале обучения, так и периодически в процессе;
- тестирование: использование диагностических и формирующих тестов для оценки уровня знаний обучаемого. Результаты тестов помогают определить, какие темы требуют дополнительного внимания и адаптации;
- анализ поведения: сбор данных о взаимодействии обучаемого с системой, таких как время, проведенное на определенных заданиях, частота выполнения упражнений и успешность выполнения. Эти данные

позволяют выявить паттерны поведения и адаптировать обучение соответственно;

- наблюдение: прямое наблюдение за обучаемыми в процессе их работы может предоставить ценную информацию о том, как они подходят к решению задач и какие стратегии используют. Этот метод может быть более трудоемким, но он позволяет получить глубокое понимание индивидуальных особенностей учащихся.

Неявные методы включают [36]:

- воплощение действий обучаемого: наблюдение за действиями игрока и интерпретация этих данных для формирования профиля обучаемого.
- отслеживание следов взаимодействия: сбор данных о действиях игрока, таких как время, проведенное на уровнях, и успешность выполнения заданий.
- устный разговор: извлечение информации о предпочтениях и мотивации обучаемого через взаимодействие с неигровыми персонажами (NPC).
- ошибки и неудачи: сбор информации о типах ошибок, совершаемых игроком, что позволяет выявить пробелы в знаниях.
- путь следования: анализ маршрута, по которому игрок проходит через игру, для понимания его предпочтений и стиля игры.

1.2.4 Методы адаптации

Методы адаптации определяют способы изменения некоторых параметров процесса обучения в зависимости от значений характеристик обучаемого, собираемых в процессе взаимодействия пользователя с системой [37].

В [38] выделены три иерархических уровня адаптации к обучаемым:

- адаптация к обучаемому как к пользователю;
- адаптация к группе обучаемых;
- адаптация к отдельному обучаемому.

На уровне адаптации к обучаемому как к пользователю уровень системы настраивается в зависимости от потребностей различных категорий пользователей. Это достигается созданием специального интерфейса, который соответствует требованиям каждой группы пользователей. Адаптация на этом уровне может включать в себя выбор тем, форматов представления информации и стилей взаимодействия.

Уровень адаптации к группе обучаемых фокусируется на характеристиках группы студентов, таких как выбранная специальность, образовательная программа, возраст и психологическая направленность личности. Адаптация к группе позволяет разрабатывать контент, который будет наиболее актуален и полезен для данной аудитории.

На уровне адаптации к отдельному обучаемому достигается максимальная степень персонализации, так как система учитывает индивидуальные характеристики каждого учащегося. Это включает в себя анализ предшествующих знаний, умений и навыков, а также личных интересов и мотивации студента. Адаптация на этом уровне позволяет создавать индивидуализированные образовательные пути, что значительно повышает эффективность обучения. Например, система может предлагать дополнительные материалы для изучения, основываясь на слабых местах обучаемого, или адаптировать темп обучения в зависимости от его прогресса.

Для адаптации к обучаемым используются различные методы, которые обеспечивают персонализированный подход в процессе обучения [39,40].

Основные методы включают:

- изменение последовательности обучения;
- адаптивное представление информации;
- интеллектуальный анализ решений;
- диалоговая поддержка решения задач;
- адаптивная поддержка в навигации.

Изменение последовательности обучения позволяет формировать индивидуально адаптированную последовательность изучения учебного

материала, что обеспечивает наиболее эффективное усвоение знаний. В компьютерных системах обучения учебный материал представляется в виде иерархии, которая может включать курсы, темы, разделы и отдельные единицы информации. Эта структура помогает управлять последовательностью изучения и позволяет создавать сценарии, соответствующие образовательным потребностям студентов.

Адаптивное представление информации ориентировано на генерацию наиболее подходящих для студента единиц информации, основанных на его предпочтениях и уровне подготовки. Современные технологии позволяют представлять учебный материал в различных форматах, таких как текст, графика и анимация. Это позволяет учитывать разные стили восприятия информации, включая визуальный, аудиальный и кинестетический;

Интеллектуальный анализ решений включает анализ ответов обучаемого с использованием интеллектуальных систем, которые способны распознавать как правильные, так и неправильные ответы. Это позволяет выявить пробелы в знаниях и предоставлять соответствующие комментарии и рекомендации для улучшения процесса обучения.

Диалоговая поддержка решения задач обеспечивает интеллектуальную помощь на каждом этапе решения задач, предлагая адаптивные комментарии и разрабатывая сценарии диалога. Сценарии диалога формируются на основе последовательности учебного материала и контрольных заданий, что позволяет эффективно оценивать достижения учащегося.

Адаптивная навигация может включать рекомендации по выбору тем и заданий в зависимости от уровня подготовки студента. Адаптивная выдача комментариев заключается в формировании комментариев в зависимости от контекста и индивидуальных особенностей пользователя [41, 42]. В таблице 1.1 представлена возможность использования методов и моделей на разных уровнях адаптации к обучаемому.

Таблица 1.1 – Методы и модели адаптации на разных уровнях

Уровень адаптации	Методы адаптации	Описание
Адаптация к пользователю	Построение интерфейса	Создание специального интерфейса для различных категорий пользователей.
	Адаптивное представление информации	Генерация контента в зависимости от предпочтений пользователя.
Адаптация к группе	Адаптация по специальности и образовательной программе	Учет характеристик группы, таких как возраст и психологическая направленность.
	Адаптивная поддержка в навигации	Рекомендации по выбору тем и заданий для группы.
Адаптация к отдельному	Интеллектуальный анализ решений	Анализ ответов студента для выявления пробелов в знаниях и предоставления рекомендаций.
	Диалоговая поддержка решения задач	Интеллектуальная помощь на каждом этапе решения задач, включая адаптивные комментарии.
	Адаптивная выдача комментариев	Формирование комментариев в зависимости от контекста и индивидуальных характеристик обучаемого.

Модель адаптации содержит адаптационный механизм, который определяет, как обновляется модель обучаемого в процессе обучения и как на основе информации из этой модели происходит адаптация информационного пространства к конкретному обучаемому. Этот механизм позволяет системе динамически изменять содержание и методы обучения в зависимости от индивидуальных характеристик учащегося.

Анализ характеристик обучаемого и механизмов адаптации в зависимости от выбранного метода адаптации представлен в таблице 1.2.

Таблица 1.2 Характеристики обучаемого и механизмы адаптации

Характеристика обучаемого	Метод адаптации	Механизм адаптации
Уровень знаний	Адаптация контента	Изменение сложности учебного материала в зависимости от уровня знаний.
Стиль обучения	Адаптивное представление	Выбор формата представления информации на основе предпочтений учащегося.
Предыдущий опыт	Интеллектуальный анализ	Анализ ответов для выявления пробелов в знаниях и предоставления рекомендаций.
Мотивация и интересы	Диалоговая поддержка	Интерактивная помощь в решении задач с учетом личных интересов и мотивации.

Ключевые компоненты модели адаптации для адаптивных обучающих систем показаны на рисунке 1.2.

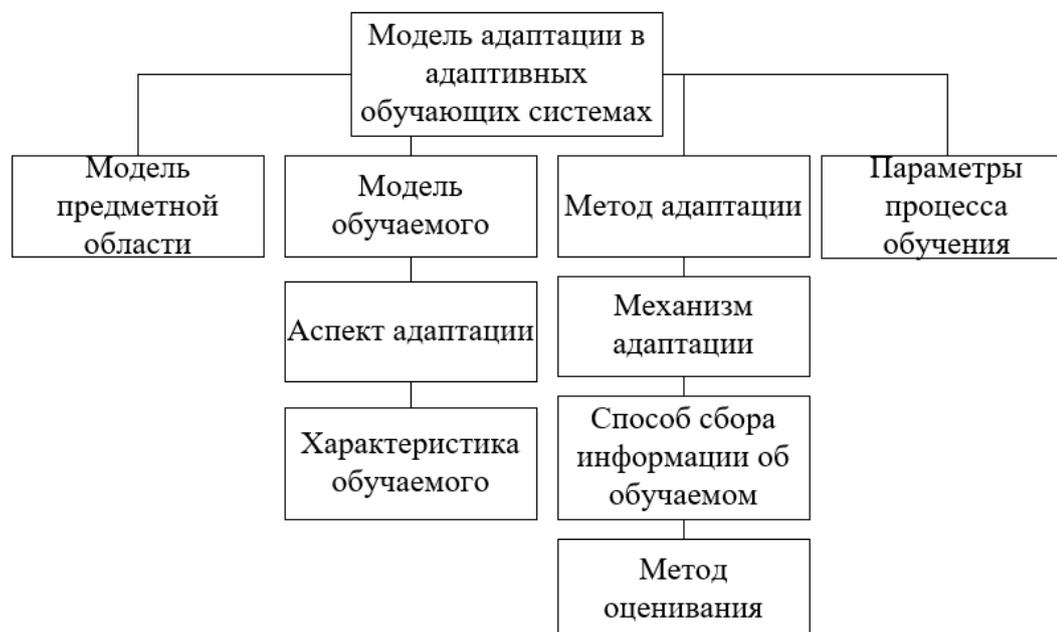


Рисунок 1.2 – Компоненты модели адаптации в адаптивных обучающих системах

1.3 Модели адаптации для адаптивных обучающих игр

Исследования в области разработки адаптивных обучающих игр развиваются по двум направлениям. Первое направление связано с использованием успешных

решений по разработке адаптивных обучающих систем для разработки обучающих игр. Однако прямой перенос моделей, методов и технологий адаптации, разработанных для адаптивных обучающих систем, в адаптивные обучающие игры может существенно ограничивать обучающие возможности и/или снижать привлекательность обучающей игры с точки зрения игровой компоненты. Поэтому в настоящее время превалирует подход, рассматривающий адаптивные обучающие игры как самостоятельную категорию ПО, требующую разработки оригинальных моделей, методов и технологий проектирования, учитывающих специфику этой категории ПО [43]. Ключевые компоненты модели адаптации для адаптивных обучающих игр показаны на рисунке 1.3.

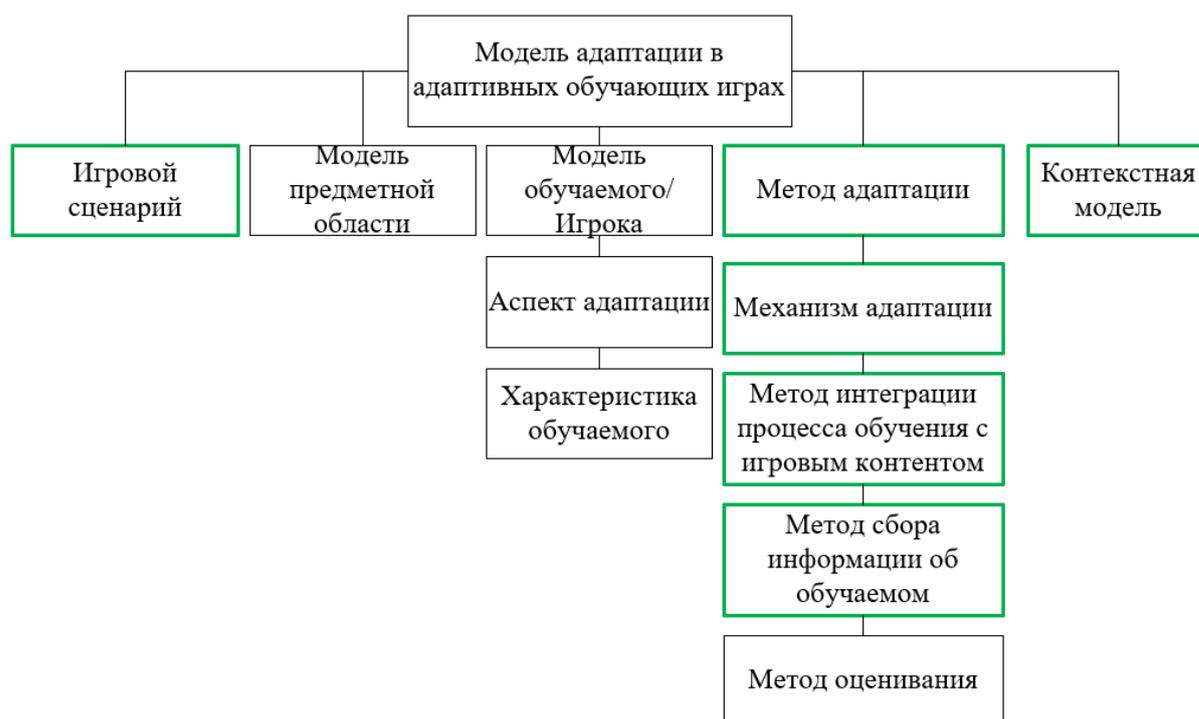


Рисунок 1.3 – Компоненты модели адаптации в адаптивных обучающих играх

Разработка моделей предметной области для адаптивных обучающих игр требует в большинстве случаев серьезных усилий экспертов при разработке модели и обновления профилей игроков [44]. В [45] предложены методы автоматического поиска моделей обучаемого для создания моделей предметной области на основе

Q -matrix метода. Q-матрица строится на основе результатов оценок тестирования студентов.

В [46] разработан метод автоматического создания модели предметной области (e-Learning domain) на основе применения технологии «Text mining» к журнальным и конференционным статьям, используемым как источники данных. Однако предложенный метод не учитывает результатов взаимодействия, обучаемых с предметной областью.

В [47] предлагается для построения модели предметной области использовать кривые обучения (learning curves). Похожий подход предложен в [48], в котором ищутся возможные пространства альтернативных моделей предметной области и находится лучшая по некоторому заданному признаку модель. Однако добавление новых концептов и отношений предметной области при этом не предусмотрено.

В наиболее поздних работах для разработки модели предметной области предлагается использование концептуальных карт (concept maps) [49] и онтологий [50].

В работе [51] предлагаются различные модели обучаемого (игрока), разработанные для адаптивных игр, включая:

- когнитивные модели (уровень знаний, скорость усвоения материала);
- эмоциональные модели (вовлеченность, уровень стресса);
- поведенческие модели (стиль взаимодействия с игрой).

Модели адаптации для адаптивных обучающих игр используют подходы на основе педагогических агентов [52-55].

В [52] предлагается модель адаптации на основе адаптивного сценария (adaptive digital storytelling), где сюжет динамически меняется в зависимости от прогресса обучаемого: например, при ошибках игрока автоматически активируются вспомогательные подсказки, а при успешном выполнении заданий открываются новые сюжетные ветки. В [56] представлен метод построения адаптивных стратегий обучения в игре, комбинирующий анализ поведения игрока в реальном времени (сбор данных о времени реакции, количестве попыток).

1.4 Встроенные модели адаптации для адаптивных обучающих игр

Для разработки адаптивных обучающих играх применяют различные модели адаптации, отличающиеся составом компонент и способами адаптации информации об обучаемом, собираемой игрой.

Адаптивная обучающая игра AutoThinking [57] предназначена для развития навыков вычислительного мышления (computational thinking (CT)). В игре реализован механизм адаптации, основанный на оценке уровня мастерства игрока и адаптивным предоставлением ему различных видов обратной связи (текстовую, графическую или видео) и подсказок. В игре реализована модель ТРМ [58] это модель прозрачного обучения которая, предоставляет обучаемым индивидуально адаптированную визуальную аналитику для изучения, включая мета когнитивные данные об образовательном прогрессе игрока (например, подсказки, предупреждения, предложения, отзывы) и метод, с помощью которого их навыки оцениваются во время игры. В модели ТРМ (рисунок 1.4), необработанные данные используются для создания различных типов аналитики процесса обучения игроков, а именно описательной, предиктивной и предписывающей аналитики. Для этого используются различные методы, в том числе методы кластеризации и классификации. Затем результаты этой аналитики визуализируются в различных формах, таких как подсказки (рекомендации и визуализация адаптивного пути обучения), предложения о потенциальных стратегических правилах, соответствующих самому последнему ходу, сделанному игроками, отзывы, предупреждения и таблицы ранжирования. В более явном виде такая визуальная аналитика, представляющая познание и компетентность игроков, и использоваться для передачи игрокам информации о приобретенных навыках и прогрессе в обучении (представляя им модель обучения игроков).

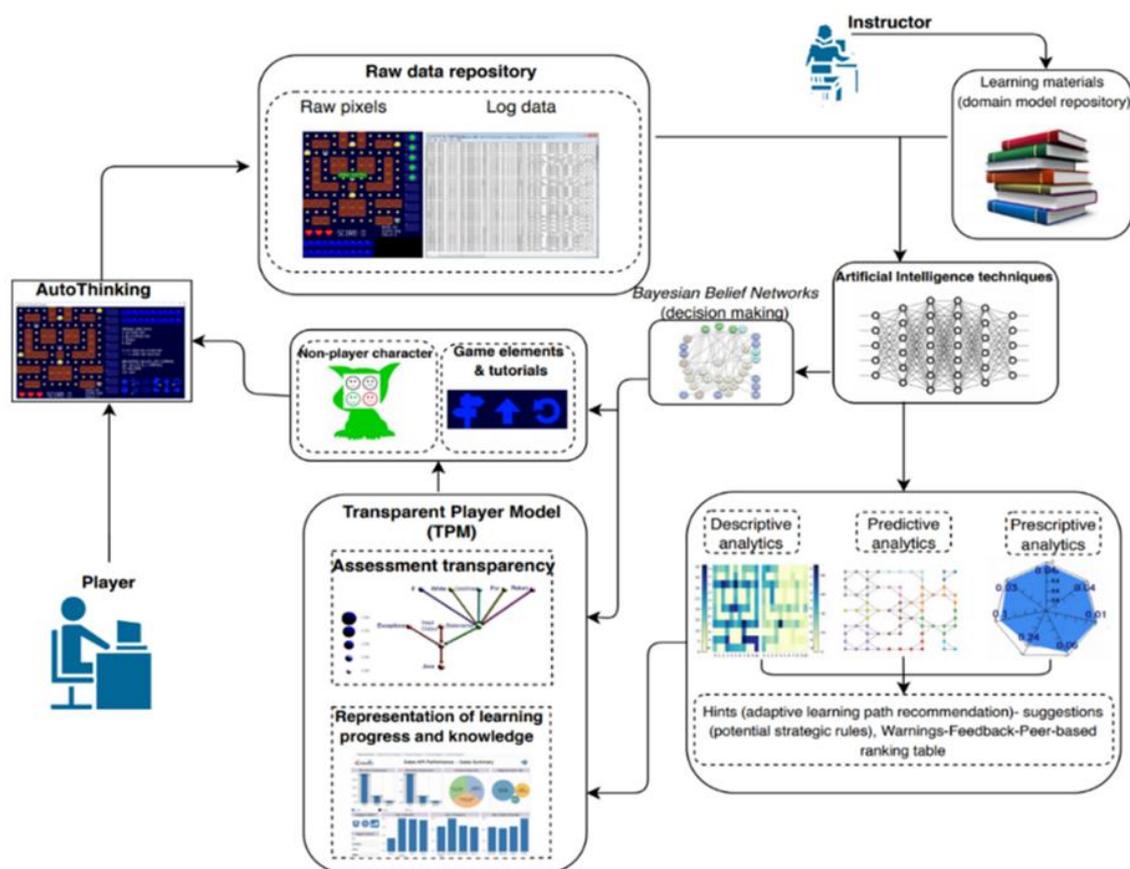


Рисунок 1.4 – Архитектура модели TPM

В ролевой приключенческой игре ELEKTRA [59] для адаптации процесса обучения используются неявный метод получения информации об обучаемом и отслеживание следов взаимодействия обучаемого с игрой. В зависимости от дальнейших действий обучаемого, модель адаптации адаптирует игру под навыки данного пользователя. В игре реализована модель адаптации ALIGN, направленная на минимизацию игрового воздействия адаптации. В основе модели лежит концептуальное разделение между игровым дизайном и обучающей адаптацией, которое осуществляется посредством процесса логического вывода, который переводит специфичность игры к абстрактным образовательным концепциям, и его реализацией в игре, переводящей абстрактные приспособления к модификациям игрового мира. Архитектура модели ALIGN (рисунок 1.5) включает четыре концептуальных процесса:

- обработки исходных данных из игры;
- построения модели пользователя;

- создание возможных приспособлений;
- применение выбранных адаптаций.

В модели ALIGN информация об обучаемом — это исходные данные, собираемые в игровом процессе. Процесс адаптации заключен в цикл, соответственно, модель адаптации на каждой итерации цикла подстраивается под вновь приобретенные знания обучаемого.

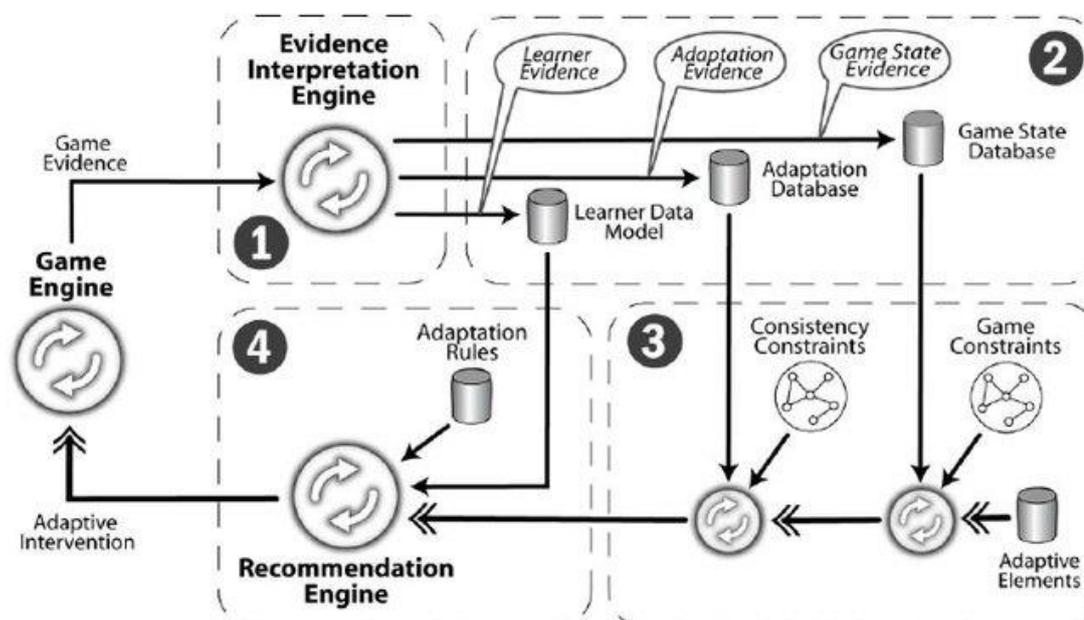


Рисунок 1.5 – Архитектура ALIGN

Примеры адаптации диалогов в игре, реализующей модель ALIGN [60] приведена на рисунках 1.6, 1.7.



Рисунок 1.6 – Экранные формы в режиме диалогов до адаптации

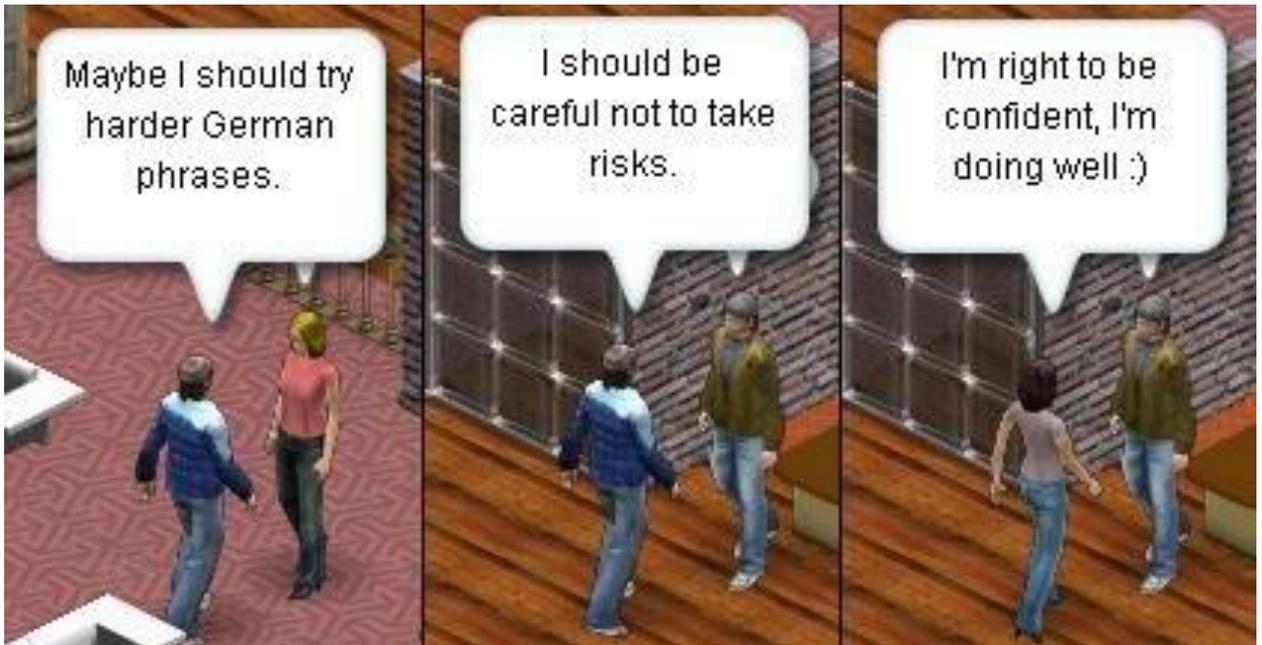


Рисунок 1.7 – Экранные формы в режиме диалогов после адаптации

В 3D ролевой приключенческой игре Code Red [61] адаптация фокусируется на уровне сложности самой игры (рисунок 1.8). Модель адаптации TRIAGE, реализованная в игре, масштабирует уровень сложности игры, используя метод «ярусной сортировки», и позволяет таким образом пользователю неявно адаптироваться к знаниям и навыкам обучаемого.

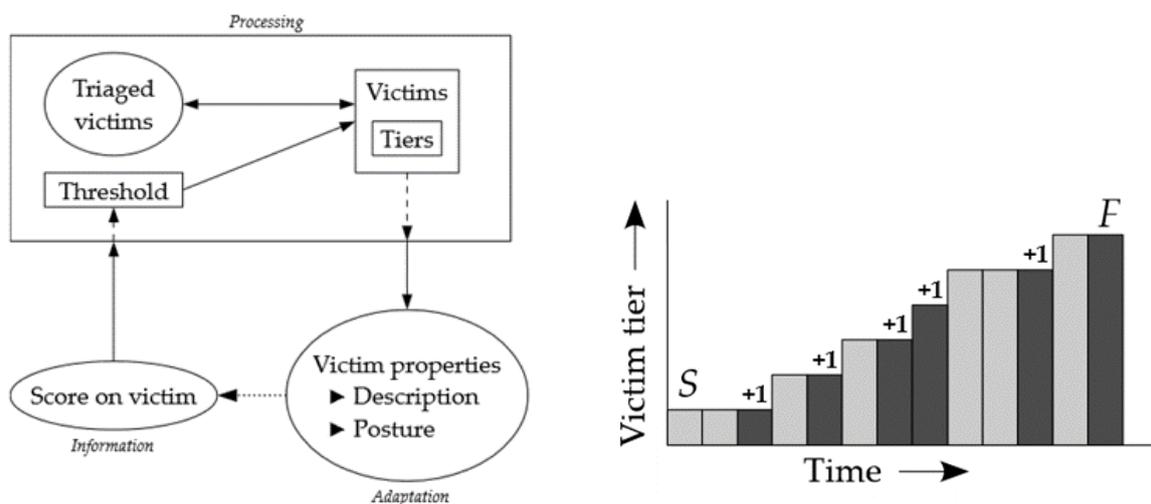


Рисунок 1.8 – Отображение процесса выбора сложности

Механизм адаптации основан на оценке психического состояния игрока, отражающего степень его вовлеченности и мотивированности, и изменения параметров процесса обучения, позволяющего достичь такого состояния (состояние потока), которое повышает эффективность обучения (рисунок 1.9).

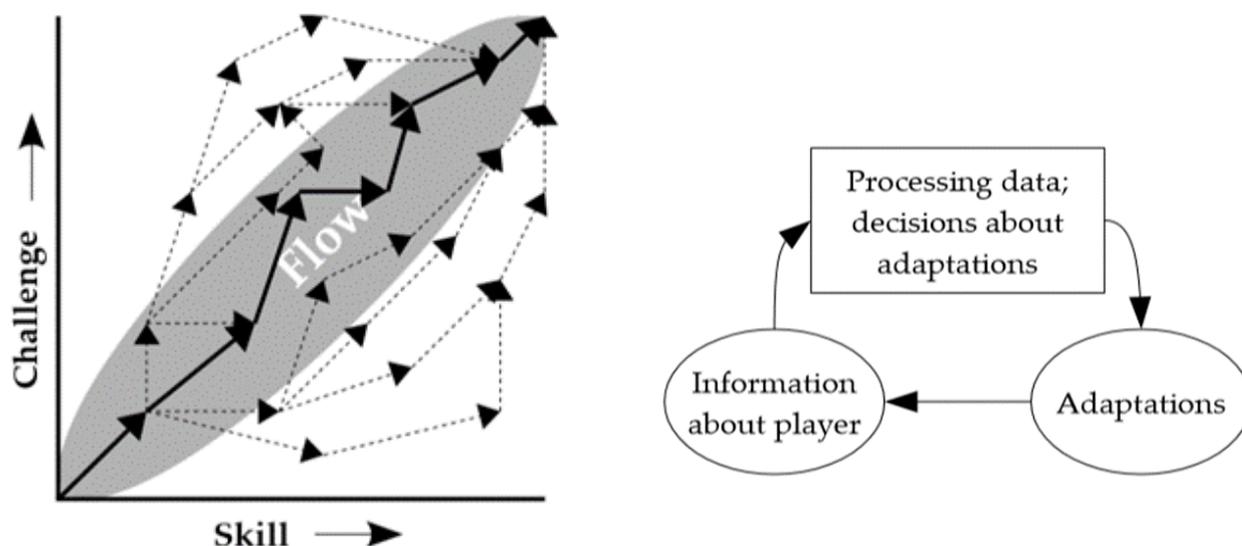


Рисунок 1.9 – Сочетание потока и контура обратной связи

В обучающей игре по программированию Minerva [62, 63] реализована модель адаптации Honey and Mumford's LSQ. Модель основана на адаптивной навигации, которая использует три уровня адаптации: запоминание, связанное с запоминанием теории; использование, связанное с применением изученной теории; и решение, связанное с решением оригинальных проблем. Для определения стратегии адаптации, порядка подачи контента, используется вводный тест, состоящий из 32 вопросов, по результатам которого определяется стиль обучаемого: теоретик, прагматик, активист, отражатель. Адаптация к стилю игры достигается за счет отслеживания предпочтений игрока и назначению ему ролей персонажей. В процессе игры стиль меняется в зависимости от действий самого игрока.

Модель адаптации ALGAE, используемая в игре Slippery Rock Falls [64], основана на объединении игрового дизайна, адаптивного обучения и учебной стратегии в одно целое. Основные компоненты игрового дизайна отражают

важность сюжета и игровой механики, а игровая деятельность выстраивается на основе конфликта. Архитектура модели ALGAE представлена на рисунке 1.10.

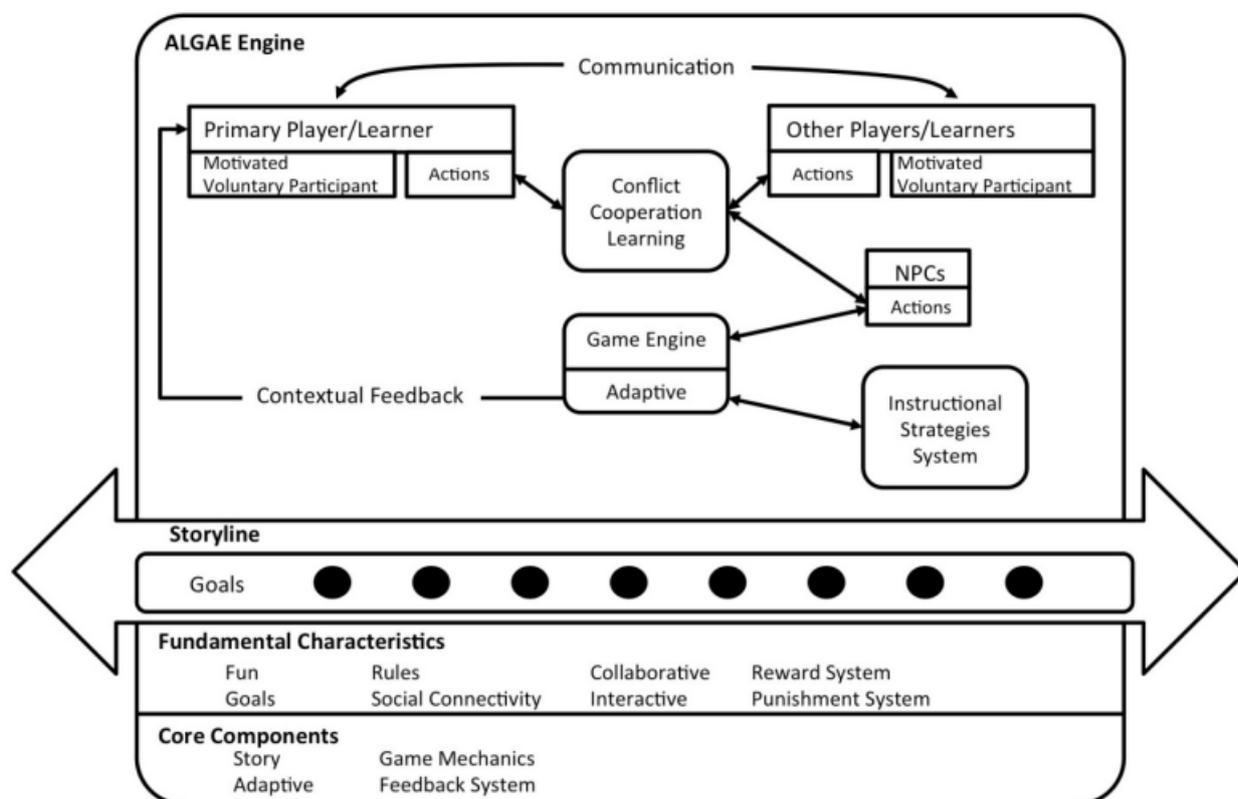


Рисунок 1.10 – Архитектура модели ALGAE

Анализ моделей адаптации для адаптивных обучающих игр представлен в таблице 1.3.

Таблица 1.3 – Анализ моделей адаптации для адаптивных обучающих игр

Модель адаптации	Аспект адаптации	Получаемая информация	Способ получения информации	Метод адаптации
TRM	Предпочтения	Действия обучаемого в процессе игры	Неявный в процессе игры	Изменение способов представления контента
ALIGN	Навыки	Действия обучаемого в процессе игры	Неявный в процессе игры	Минимизация игрового воздействия

Продолжение таблицы 1.3

Модель адаптации	Аспект адаптации	Получаемая информация	Способ получения информации	Метод адаптации
Honey and Mumford's LSQ	Стиль и предпочтения	1. Действия обучаемого в процессе игры 2. Ответы на анкетирование: входящее, промежуточное и финальное	1. Неявный, в процессе игры 2. Явный, с помощью вопросника	1. Изменение уровня сложности 2. Изменения типов подсказок
TRIGE	Знания и навыки	1. Действия обучаемого 2. Ответы обучаемого на задания	Неявный в процессе игры	Изменение уровня сложности
ALGAE	Навыки и предпочтения	Действия обучаемого	Неявный в процессе игры	Изменения типов подсказок

В современных адаптивных обучающих играх преобладают специализированные системы персонализации, которые создаются под конкретный игровой проект и не обладают универсальностью. На сегодняшний день одной из ключевых разработок в этой области считается платформа ALGAE, сочетающая методы адаптивного обучения с игровыми механиками. Несмотря на её потенциал для применения в образовательных и исследовательских целях, реализация данной системы ограничена закрытой архитектурой оригинальной игровой среды, что исключает её адаптацию для сторонних проектов без глубокой модификации исходного кода. Это подчеркивает актуальность задачи создания гибких кроссплатформенных решений, способных масштабироваться для различных жанров обучающих игр.

1.5 Инструменты проектирования адаптивного обучающего ПО

Для разработки адаптивного обучающего ПО применяют такие инструменты специализированные программные средства, которые различаются функциональными возможностями и применяемыми моделями адаптации.

EdApp [65,66] это платформа для разработки адаптивных обучающих систем, позволяющая создавать самому или использовать готовые решения поддерживающие элементы геймификации. Платформа реализует обучение методом интервального повторения, устраняя проблему сохранения знаний. Благодаря автоматическим персонализированным тестам Brain Boost позволяет учащимся лучше запоминать новую информацию. Тесты позволяют больше сосредоточиться на заданиях, которые выполнены неправильно, осуществляя персонализацию для каждого учащегося, в каждой новой сессии на основе ответов из предыдущих сессий.

ALE [67,68] это среда электронного обучения для разработки адаптивных обучающих систем, которая позволяет подстраиваться под стиль обучения за счет модели VARK. Согласно модели VARK, учащиеся делятся на четыре группы, представляющие основные стили обучения, на основе их ответов, где каждый ответ соответствует одному из крайних значений измерения, которые используют его для создания персонализированных обучающих курсов.

ALOSI [68,70] это платформа адаптивного обучения с открытым исходным кодом, разработанная Гарвардским университетом и Microsoft в партнерстве с edX. Предлагаемое решение состоит из адаптивного движка и моста для адаптации, системы управления обучением и контентом. Эта структура использует и то, и другое для моделирования состояния знаний учащихся.

Canvas LMS [71] это система управления обучением с открытым исходным кодом, которая предоставляет возможность разработки курсов в соответствии с разными моделями обучения. Кроме того, Canvas предлагает функцию MasteryPaths, которая позволяет создавать дифференцированные задания, и перенаправлять обучаемых на разные пути обучения.

Protus [72] – это веб-система обучения, позволяющая выстраивать два стратегии персонализации. Первый основан на адаптивных гипермедиа, а второй на методах рекомендательных систем. Адаптивные гипермедиа помогают студенту в ходе курса, предлагая страницы, соответствующие требованиям учащегося. Система рекомендаций построена на взаимодействие с учащимся, в зависимости от оценки за задания, система строит рекомендации на основе онтологий и выполняет настройку пользовательского интерфейса.

Moodle [73] – это система для построения персонифицированных стратегий. Предлагаемая система основана на гибридной модели, которая использует автоматическое определение стиля обучения и опросник Index of Learning Styles (ILS) для повышения точности. Данный подход позволяет адаптироваться к стилю обучения учащихся с использованием методов интеллектуального анализа данных.

Concerto [74] – это платформа с открытым исходным кодом, разработанная центром психометрии Кембриджского университета и используемую для адаптивных оценок. Это наиболее известная платформа, предназначенная для обеспечения веб-применения подхода CAT. Concerto использует язык статистических вычислений для создания персонализированного теста, он разработан с учетом способностей каждого профиля учащегося с помощью специальных тестов.

Mod-Knowledge agent [75, 76] – это мультиагентная система предоставляющая возможность изменять алгоритм, используемый каждым агентом, в зависимости от внешней среды. Система включает в себя агент под названием "Mod-Knowledge", который отслеживает состояние знаний с учетом внешней навигации учащегося. В этом агенте используется алгоритм DKVMN, который модифицирован в соответствии с контекстом платформы электронного обучения. Модель DKVMN является улучшением модели ДКТ. Кроме того, алгоритм может обнаруживать учащихся, испытывающих трудности, с которыми они сталкиваются в ходе продвижения обучения.

SCOPE-IT [77, 78] – это система разговорного обучения, которая использует объекты знаний в качестве портативных интеллектуальных наставников. Система может быть встроена в существующую стратегию обучения, а разработанный метод позволяет аннотировать HTML-страницы с обучающими диалогами на естественном языке, которые объясняют и обсуждают содержимое в каждом блоке страницы. Используемые данные были собраны из коммерческой адаптивной системы ALEKS, системных журналов, предпочтений учащихся, способностей учащихся и анкетного теста, основанного на математических концепциях.

Однако перечисленные инструменты (таблица 1.4) не ориентированы на разработку адаптивных обучающих игр. Одним из последних развивающихся решений в сфере разработки адаптивных обучающих игр для является система GLA [79, 80], которая внедряется на этапе разработке игры. Система реализует оценку результатов обучаемого в ходе прохождения игры и позволяет отображать результаты с помощью адаптивного интерфейса. Однако применение системы GLA для разработки адаптивных игр требует самостоятельной разработки модели адаптации и ее ключевых компонент.

Таблице 1.4 – Анализ средств разработки адаптивных обучающих систем

Название	Метод	Модель адаптации	Характеристики обучаемого
EdApp	Интеллектуальный анализ решений	Brain Boost	Уровень знаний
ALE	Адаптивное представление информации; Построение последовательности обучения	VARK	Стиль обучения; Уровень знаний
ALOSI	Построение последовательности обучения	edX	Уровень знаний
Canvas LMS	Построение последовательности обучения	MasteryPaths	Уровень знаний

Продолжение таблицы 1.4

Название	Метод	Модель адаптации	Характеристики обучаемого
Protus	Адаптивная поддержка в навигации	Онтология	Предпочтения
Moodle	Адаптивное представление информации; Построение последовательности обучения	ILS	Стиль обучения; Уровень знаний
Mod-Knowledge agent	Адаптивная поддержка в навигации	DKVMN	Предпочтения
SCOPE-IT	Адаптивное представление информации	ALEKS	Способ восприятия
Concerto	Адаптивное представление информации	CAT	Стиль обучения

1.6 Выводы по главе

В Главе 1 рассмотрены научно-методические основы проектирования адаптивного обучающего программного обеспечения. Проведен анализ современных тенденций в развитии обучающих систем, включающих адаптивные обучающие системы, персонифицированные процессы обучения, обучающие игры — как самостоятельные игровые формы, так и адаптивные обучающие игры, сочетающие элементы адаптивных систем и компьютерных игр. В рамках исследования рассмотрены следующие вопросы:

- модели предметной области, представляемые графами общего вида, задающими конечное множество фрагментов содержания области знаний и связи между ними, отражающие логику освоения материала;
- модели обучаемого, предназначенные для хранения характеристик пользователя, учитываемых при адаптации, и определяющие аспекты процесса адаптации;

- методы сбора информации об обучаемом, включающие явные и неявные подходы, направленные на получение данных о знаниях, навыках и поведении пользователя;
- методы адаптации, определяющие способы изменения параметров процесса обучения на основе мониторинга состояния обучаемого в ходе взаимодействия с системой.

Показано, что разработка модели адаптации является наиболее сложным и трудоемким этапом проектирования адаптивных обучающих игр, требующим высокой квалификации разработчика и существенно влияющим на качество конечного продукта. Известные модели адаптации, созданные специально для адаптивных обучающих игр, являются встроенными ad-hoc моделями, привязанными к конкретной реализации одной игры. Одним из перспективных решений в данной области является система GLA, внедряемая на этапе разработки игры. Однако использование системы GLA требует самостоятельной разработки модели адаптации, что повышает сложность процесса.

Инструменты, предназначенные для разработки адаптивного обучающего программного обеспечения, не ориентированы специально на создание адаптивных обучающих игр. Для повышения качества и снижения трудоемкости проектирования таких игр необходимо разработать переносимую модель адаптации, применимую к различным жанрам и типам игр, а также технологию автоматизированного проектирования адаптивных обучающих игр на основе этой модели.

Глава 2. Разработка переносимой модели адаптации для адаптивных обучающих игр

Для разработки модели адаптации применен подход к моделированию процесса обучения как динамики состояния обучаемого на пространстве знаний, и исчислением состояний на этом пространстве в идемпотентной алгебре.

В качестве основы выбрана модель [81], которая рассматривает процесс обучения как процесс взаимодействия обучаемого с предметной областью. Соответственно, модель процесса обучения представлена тремя моделями: моделью предметной области, моделью обучаемого и моделью взаимодействия обучаемого с предметной областью. Процесс обучения рассматривается как управляемый процесс взаимодействия обучаемого с обучающим курсом. Предметная область рассматривается как пространство знаний, образуемое множеством элементов, как фрагментов знаний, объединённых в целое отношениями между ними, сохраняющими логику целого. Пространство знаний, рассматриваемая как система, обладает свойствами целостности и делимости. Целостность пространства знаний означает принципиальную несводимость знания в целом к сумме фрагментов знаний. Делимость пространства знаний означает наличие в нем подпространств, обладающих свойствами пространства.

Взаимодействие обучаемого с обучающим курсом представляется как освоение обучаемым элементов пространства знаний и логических связей между элементами, определяемых свойствами этого пространства. Процесс освоения представляет собой выполнение действий по освоению обучаемым пространства знаний, как системно организованной структуры. Освоение пространства знаний подразумевает освоение всех элементов этого пространства в их логической взаимосвязи.

Освоение элемента пространства означает выполнение обучаемым действия, предписанного разработчиком курса этому элементу. В результате выполнения каждого действия обучаемый приобретает новые знания и/или развивает навыки. Таким образом, после выполнения каждого действия обучаемый изменяет свое

состояние освоенности пространства знаний. Состояние освоенности обучаемым пространства знаний определяется как суперпозиция значений состояний освоенности всех элементов пространства знаний.

Обучаемый, в процессе освоения пространства знаний, выполняет действия, определяемые элементами пространства в их логической связности. При этом выбор последующего действия обучаемого определяется совокупностью всех выполненных действий, обеспечивающих возможность выполнения данного действия. Всякое действие обучаемого состоит в том, что оно имеет направленность, связанную с изменением состояния освоенности. Всякому состоянию обучаемого отвечает подпространство освоенных знаний. Обучаемый совершает такие действия, которые расширяют это подпространство, и, в конечном итоге, покрывают все пространство.

Таким образом, процесс обучения представляется как деятельность обучаемого по освоению пространства знаний как системно организованной структуры, приводящая к интеграции знаний, полученных в результате выполнения действий, и приводящая к синтезу целостной системы знаний у обучаемого.

Такой подход к представлению процесса обучения требует выделения трех компонент процесса: обучающего курса, обучаемого, и процесса взаимодействия между обучаемым и обучающим курсом (рисунок 2.1).

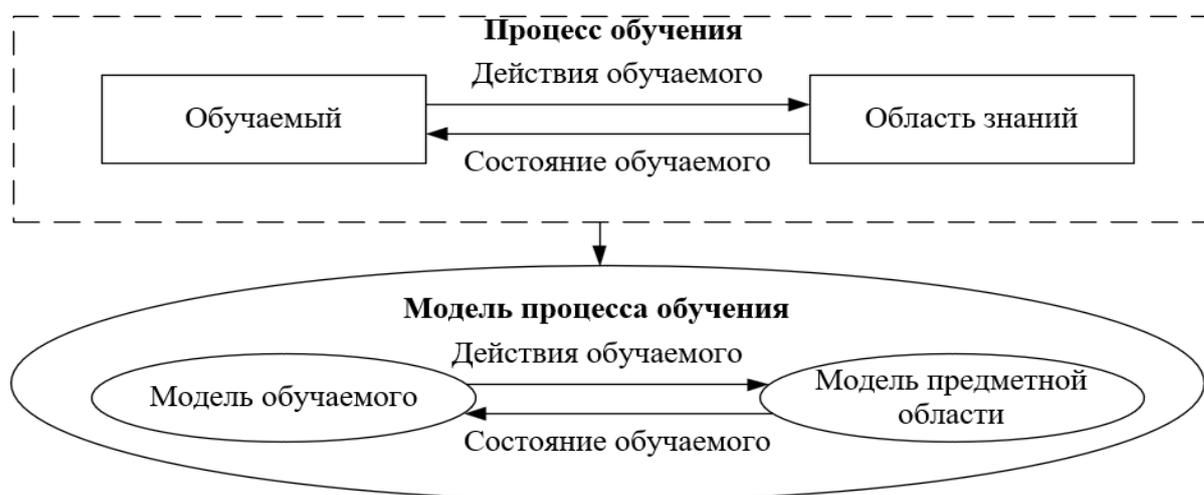


Рисунок 2.1 – Моделирование процесса обучения

2.1 Модель предметной области

Обучающий курс представляется набором связанных элементов. Каждому элементу структуры сопоставляется фрагмент обучающего курса. В качестве элементов курса могут выступать блоки теоретического материала (лекции, и/или их фрагменты), практические задания, тесты различного вида, упражнения и т.д. Связи между элементами задают логическую связность курса, отражающую его внутреннее устройство [82].

В общем случае обучающий курс можно описать как конечное множество с заданным на нем бинарным отношением (формула 2.1)

$$KnowledgeStructure = \langle Concepts, \leq \rangle, \quad (2.1)$$

где $Concepts = \{c_1, \dots, c_i, \dots, c_n\}$ – множество фрагментов знаний, \leq – бинарные отношения между фрагментами, отражающие логику их усвоения.

Отношение \leq означает, что два элемента связаны между собой с точки зрения последовательности изучения. Так, элементы c_1 и c_2 связаны бинарным отношением $c_1 \leq c_2$, если освоение элемента a является с точки зрения разработчика курса необходимым для освоения элемента c_2 , т.е. c_1 является основанием для c_2 . Если допустить, что при этом также справедливо отношение $c_2 \leq c_1$, т.е. освоение элемента b является с точки зрения разработчика курса необходимым для освоения элемента c_1 , то это означает, что c_1 и c_2 один и тот же элемент.

При построении обучающего курса разработчик выделяет элементы курса и задает попарно связи между ними. Логическая связность всего курса не допускает появления замыкания связей (циклов) в пространстве знаний, т.е. одновременное существование отношений вида $c_3 \geq c_2$, $c_1 \geq c_3$ недопустимо.

Для исключения избыточности обучающего курса отношение \leq должно обладать свойством антисимметрии. Для обеспечения логической связности элементов курса отношение \leq не должно допускать последовательности элементов, образующих цикл.

Таким образом, логическая связность, как бинарное отношение, обладает следующими свойствами:

- элементы c_1 и c_2 логически связаны отношением $c_1 \leq c_2$, если освоение a является с точки зрения разработчика курса необходимым для освоения c_2 , т.е. c_1 является основанием для c_2 ;
- никакой элемент не может быть опосредованно основанием самого себя, т.е. набор отношений $c_1 \leq c_2$ и $c_2 \leq c_3$ и $c_3 \leq c_1$ недопустим;
- каждый элемент курса является основанием для всех элементов, с ним связанных, и для всего курса в целом, т.е. если $c_1 \leq c_2$ и $c_2 \leq c_3$, то $c_1 \leq c_3$ (отношение \leq является транзитивным).
- свойства логической связности позволяют определить множество как упорядоченное множество, и отношение \leq как отношение порядка на нем.

Графовое представление (рисунок 2.2), как наиболее общее представление бинарных отношений, достаточно для моделирования структуры предметной области [83, 84]. Но для организации процесса освоения знаний необходимо, чтобы модель предметной области позволяла вычислять знания, накопленные каждым обучающимся при освоении предметной области. В рамках предложенного подхода [85] модель предметной области представляется не метрическим пространством знаний (рисунок 2.3), образуемым вложением множества фрагментов знаний $C = \{c_1, \dots, c_i, \dots, c_n\}$, где c_i – фрагмент знаний, связанных логикой освоения, в решетку как алгебраическую структуру (формула 2.2):

$$KS = \langle S, (*, \oplus) \rangle, \quad (2.2)$$

где KS – пространство знаний (Knowledge Space), $S \supset C$ – множество элементов пространства, $(*, \oplus)$ – бинарные операции на пространстве.

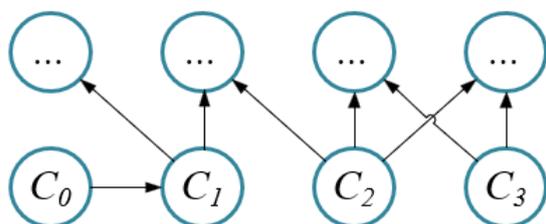


Рисунок 2.2 – Графовое представление предметной области

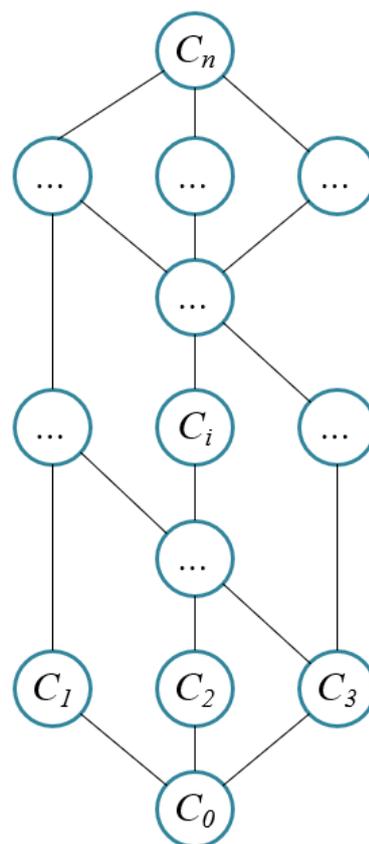


Рисунок 2.3 – Пространство знаний

Каждому фрагменту знаний c_i сопоставляется действие la_i (learning activities), требуемое для освоения этого фрагмента (например, изучение теоретических материалов, выполнение практического задания, прохождение тестирования и т.д.). Каждому действию, выполненному обучаемым в процессе освоения пространства, сопоставляется оценка освоенности соответствующего фрагмента знаний, определяющая состояние обучаемого $\varphi(c_i)$. Уровень освоенности рассчитывается на основе объективных метрик или субъективных данных:

- точность выполнения заданий (доля верных ответов);
- время выполнения;
- количество попыток;
- самооценка обучаемого (анкеты);
- анализ вовлеченности (клики, активность в обсуждениях).

При этом пространство действий (рисунок 2.4) и пространство состояний (рисунок 2.5) структурно-эквивалентны пространству знаний.

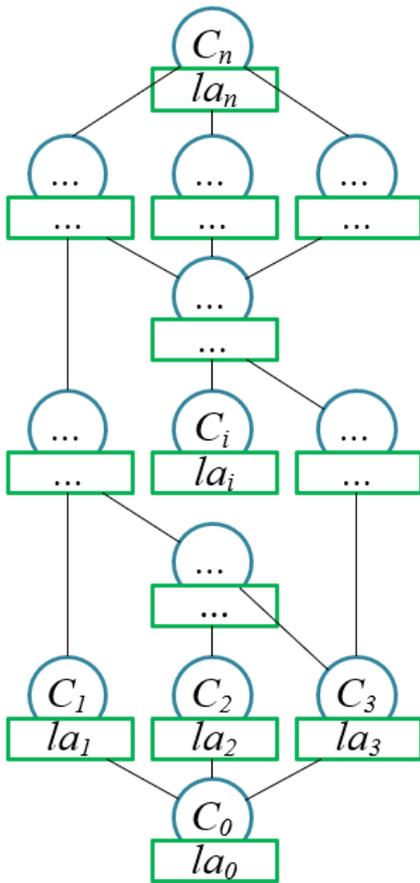


Рисунок 2.4 – Пространство действий

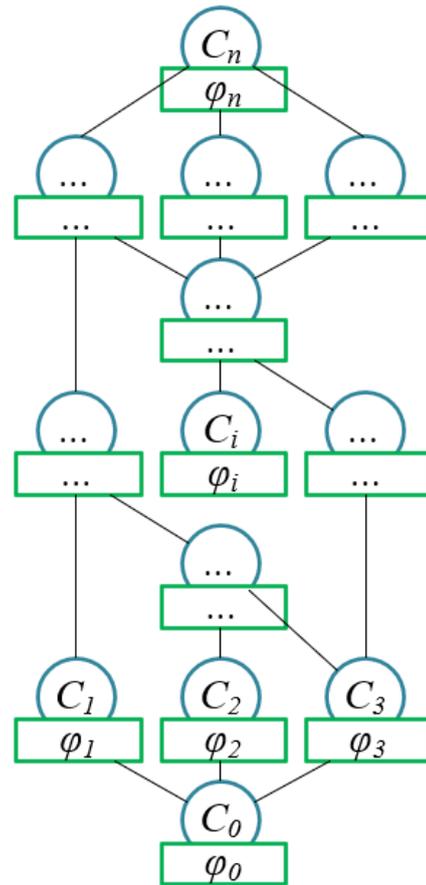


Рисунок 2.5 – Пространство состояний

Анализ подходов к интеграции обучающих и игровых компонентов позволяет выделить четыре способа организации сценариев в обучающих играх [86, 87]:

- обучающий сценарий;
- обучающий сценарий с элементами игры;
- независимый игровой и обучающий сценарий;
- игровой сценарий с элементами обучения.

Под сценарием обучающей игры понимают набор взаимосвязанных элементов сценария, представляющих игровую и обучающую компоненты. Доминирование в игре обучающей или игровой компоненты определяет класс обучающей игры (рисунок 2.6).

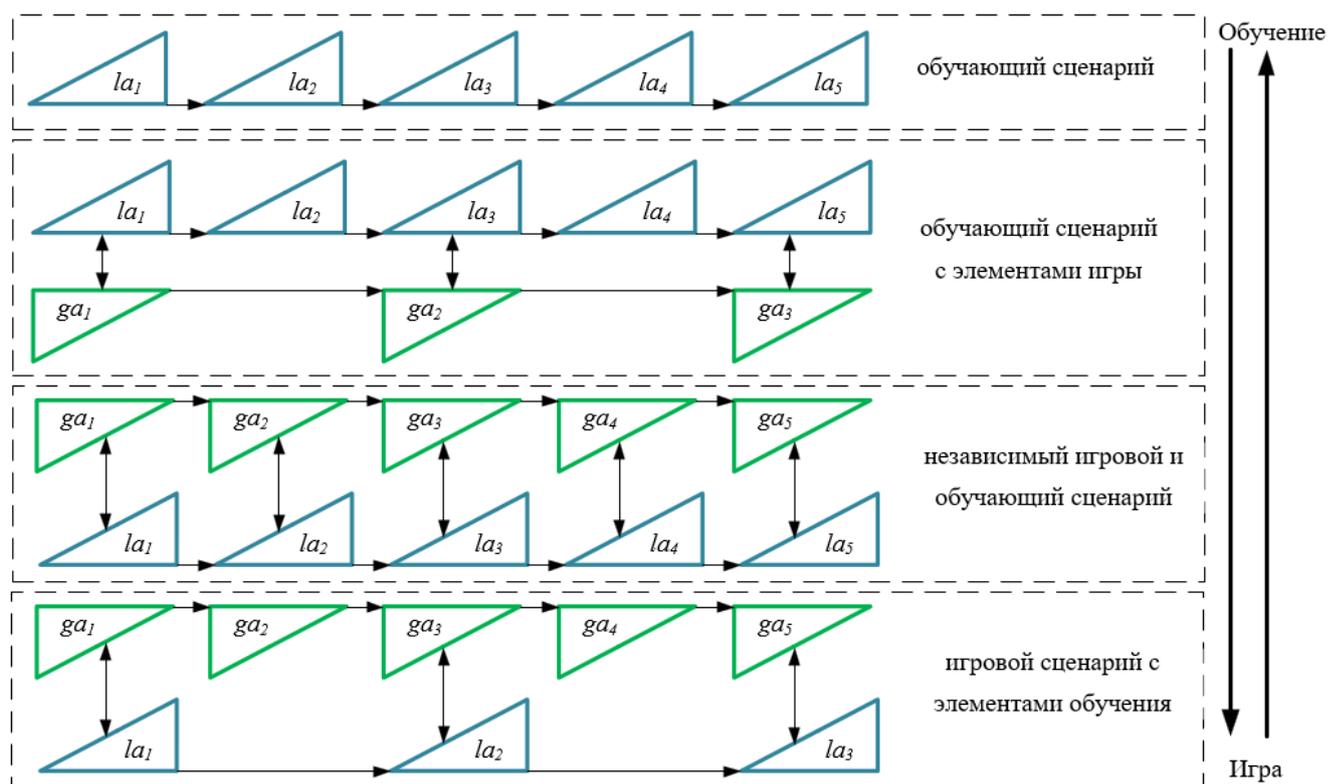


Рисунок 2.6 – Способы организации сценариев в обучающих играх

Типичными примерами игр, в которых реализован только обучающий сценарий, являются игры-симуляторы. В ряде работ такие игры выделяют в отдельную категорию обучающих систем и не относят к категории игр. Игровая составляющая в таких играх реализуется с использованием богатых графических и звуковых возможностей современных средств вычислительной техники для компьютерной интерпретации реальных процессов.

Применение в обучающих системах различных игровых элементов определяет принадлежность таких систем к обучающим играм второй группы.

Третий способ организации сценария основан на разработке обучающих игр, в которых существуют два непересекающихся сценария, и, соответственно, две непересекающиеся цели – цель обучения и игровая цель. Такой способ организации сохраняет игровую привлекательность, однако не гарантирует достижения обучающей цели. Интеграция отдельных обучающих действий в игровой сценарий позволяет усилить обучающий эффект, однако цель обучения в явном виде также не определяется.

Для обеспечения одновременного достижения игровой и цели обучения необходимо, что игровые и обучающие задания выполняются одновременно, т.е. объединены в одно действие (рисунок 2.7). Такая модель сценария (комбинированная модель) позволяет обеспечивать баланс игровой и обучающей компоненты сохранять, таким образом, целостность восприятия игры, и достигать цели обучения в результате достижения игровой цели, т.е. такая модель реализует принцип *quiproquo*. Игрок стремится к достижению игровой цели, но при этом он неявно стремится к достижению цели обучения, т.е. игровая цель достигается как цель обучения [88].



Рисунок 2.7 – Комбинированная модель сценария

В [89] разработана концепция погружения процесса обучения, в игровой сценарий, которая обеспечивает баланс игровой и обучающей компоненты и обеспечивает возможность достижения цели обучения в результате достижения игровой цели. Модель погружения основана на сопряжении процесса обучения и игрового процесса. Каждому компоненту процесса обучения сопоставляется его эквивалентная интерпретация в игровом контексте, что позволяет обеспечивать структурное единство процесса обучения и игрового процесса и эквивалентность достижения целей обучения и самой игры. Интеграция обучающих действий в игровой контекст основана на реализации трех I-компонентов (3I-approach) [90–92]. Материалы курса представляются (Introduction) через игровой мир; игровые задания, выполняемых игроком, и их решения интерпретируются в контексте игры (Interpretation); результаты изучения курса влияют (Influence) на результаты игры (рисунок 2.8).

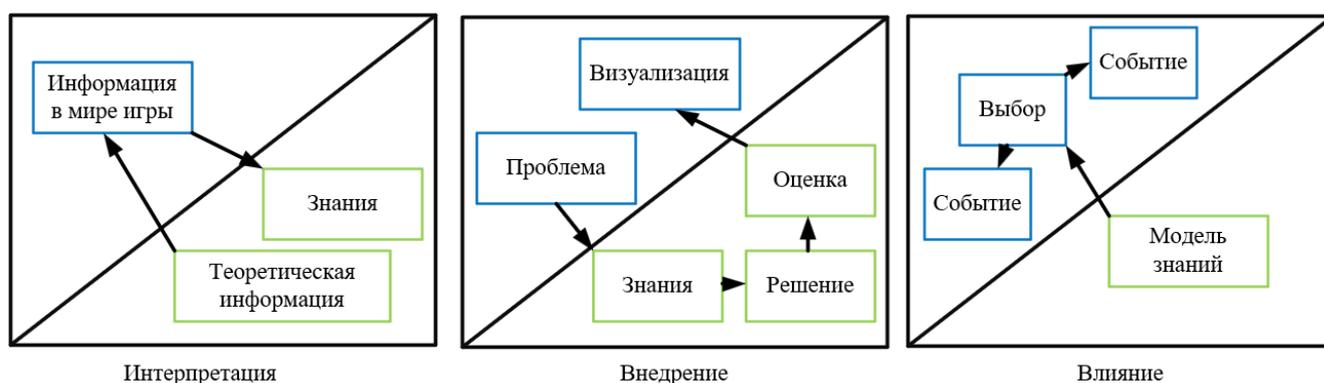


Рисунок 2.8 – Интерпретация компонентов модели обучающего курса

Для обеспечения одновременного достижения игровой и цели обучения предложена комбинированная модель сценария, в котором игровые и обучающие задания выполняются одновременно, т.е. объединены в одно действие (рисунок 2.9).

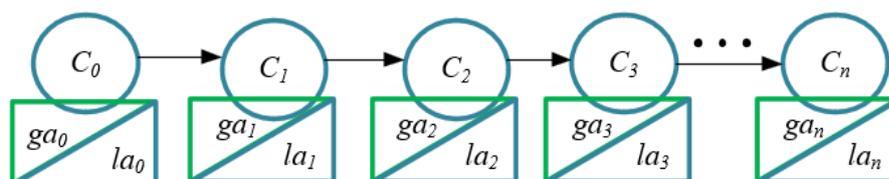


Рисунок 2.9 – Модель линейного комбинированного сценария

Предложенная модель сценария реализует принцип *quiproquo*. Игрок стремится к достижению игровой цели, но при этом он неявно стремится к достижению цели обучения, т.е. игровая цель достигается как цель обучения.

Проблемой разработки адаптивных обучающих игр с комбинированным нелинейным сценарием, реализующих предложенную модель погружения обучающего сценария в игровой контекст на основе 3I-подхода, является необходимость разработки способа интеграции процесса обучения с игровым процессом, сохраняющего логику нелинейных обучающего и игрового сценариев. Модель нелинейного комбинированного сценария представлена на Рисунке 2.10.

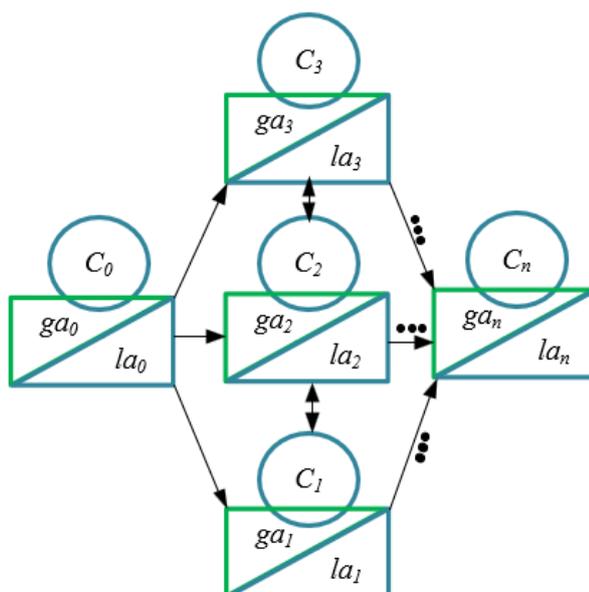


Рисунок 2.10 – Модель нелинейного комбинированного сценария

2.2 Модель обучаемого

Модель обучаемого представляет собой кортеж вида (формула 2.3) [93, 94]:

$$Learner = \langle \Phi, LA \rangle, \quad (2.3)$$

где $\Phi = \{\varphi_1, \dots, \varphi_i, \dots, \varphi_m\}$ – множество последовательных состояний обучаемого; $LA = \{la_1, \dots, la_i, \dots, la_m\}$ – множество последовательных действий обучаемого.

Структурная упорядоченность пространства позволяет представлять процесс освоения пространства знаний как зависимость состояния обучаемого на пространстве от действий, сопоставленных элементам пространства, и изменяющих состояние обучаемого на этом пространстве. При этом каждый обучаемый свободен в выборе действий по освоению пространства знаний, определяемых его текущим состоянием и структурой самого пространства. Модель процесса обучения представляется уравнением эволюции состояния обучаемого в пространстве состояний (формула 2.4):

$$\varphi_{i+1} = la_{i+1}(\varphi_i), \quad (2.4)$$

где φ_{i+1} – состояние, в которое переходит обучаемый после выполнения действия la_{i+1} , определяемого предшествующим состоянием φ_i ; $\varphi \in \Phi$, $la \in LA$, где Φ и LA определяются моделью обучаемого.

Любая стратегия обучения, формируемая обучаемым, включает все элементы пространства знаний, но стратегии различаются порядком их следования, формируемым самим обучаемым. Примеры возможных стратегий освоения пространства знаний показаны на рисунке 2.11.

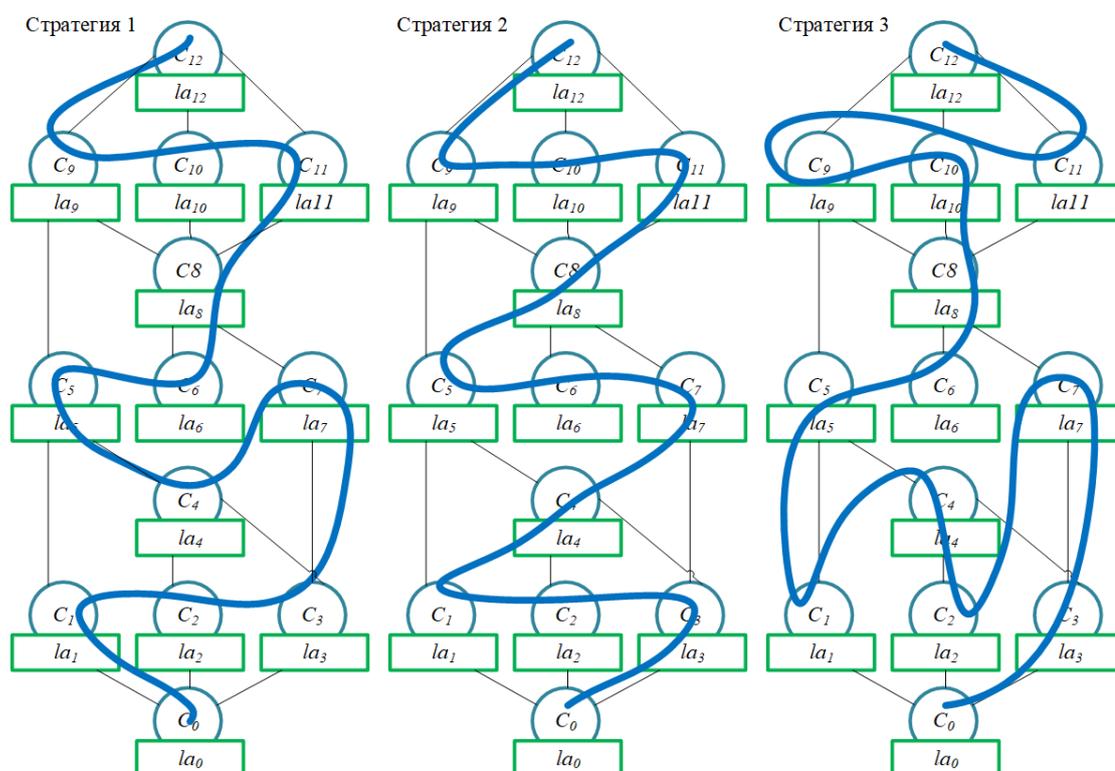


Рисунок 2.11 – Пример возможных стратегий освоения пространства знаний

2.3 Метод адаптации

Метод адаптации процесса обучения к уровню знаний обучаемого основан на свободном формировании обучаемым стратегии по освоению пространства знаний в игровом контексте, определяемой текущим состоянием обучаемого и структурой самого пространства. При этом любая игровая стратегия обучения, формируемая обучаемым, включает все элементы пространства знаний, но

стратегии различаются порядком их следования, формируемым самим обучаемым. Разработанный метод адаптации обеспечивает свободу выбора игровых действий, не противоречащих логике процесса обучения. Таким образом, стратегия обучения, формируемая на пространстве действий в игровом контексте, является адаптивной как с точки зрения игрового процесса, так и с точки зрения процесса обучения (рисунок 2.12).

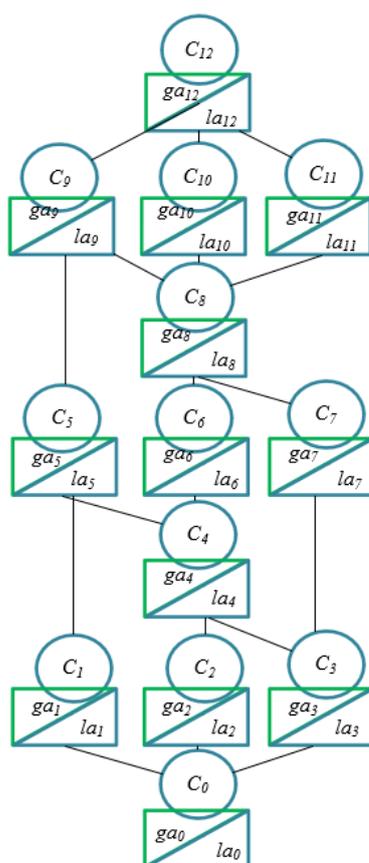


Рисунок 2.12 – Пространство знаний в игровом контексте

Пространство знаний, как нелинейная структура, может содержать множества фрагментов знаний ($c_1, \dots, c_i, \dots, c_n$), не связанных логикой их освоения. Порядок освоения таких фрагментов выбирает обучаемый. Но так как в игровом сценарии каждое действие обучаемого la_i интерпретировано как игровое действие ga_i , то любой выбранный обучаемым порядок действий должен следовать логике игрового сценария. Разработан метод адаптации процесса обучения в обучающих играх со сценарием, основанный на динамическом контентном согласовании нелинейных обучающего и игрового сценариев. Для обеспечения динамической

совместимости обучающего и игрового сценариев, любой последовательности независимых обучающих действий ($la_1, \dots, la_i, \dots, la_n$), выбираемых игроком в процессе игры, генерируются последовательности из сопоставленных им игровых действий ($ga_1, \dots, ga_i, \dots, ga_n$), образующие связные фрагменты игрового сценария (Рисунок 2.13).

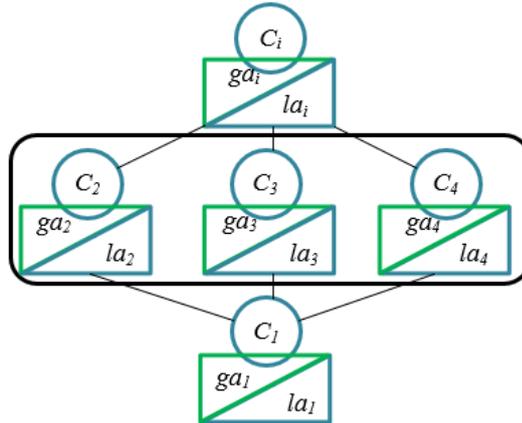


Рисунок 2.13 – Фрагмент пространства действий

В общем случае, для последовательности из n независимых обучающих действий возможны $n!$ вариантов последовательностей игровых действий в игровом сценарии. Пример реализации игрового сценария приведен на Рисунке 2.14.

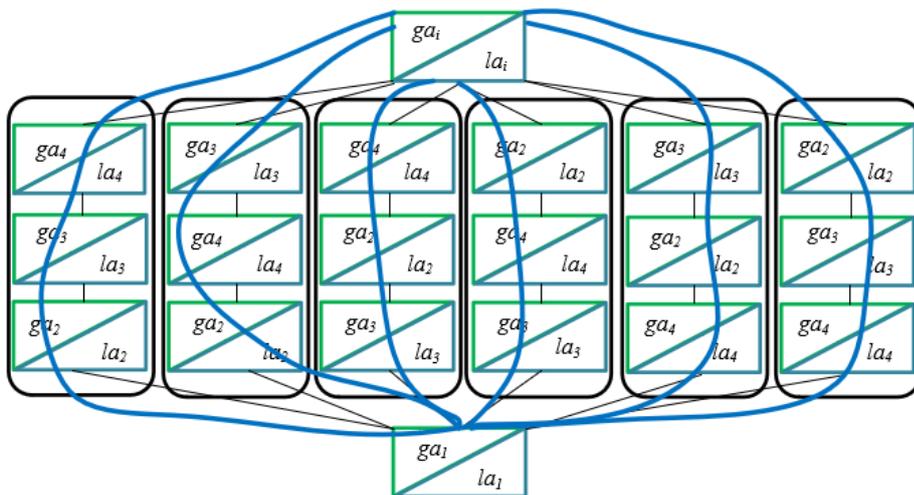


Рисунок 2.14 – Фрагмент игрового сценария, сопоставленного фрагменту пространства знаний

Примеры возможных стратегий освоения фрагмента игрового сценария, сопоставленного фрагменту пространства знаний представлены на рисунке 2.15.

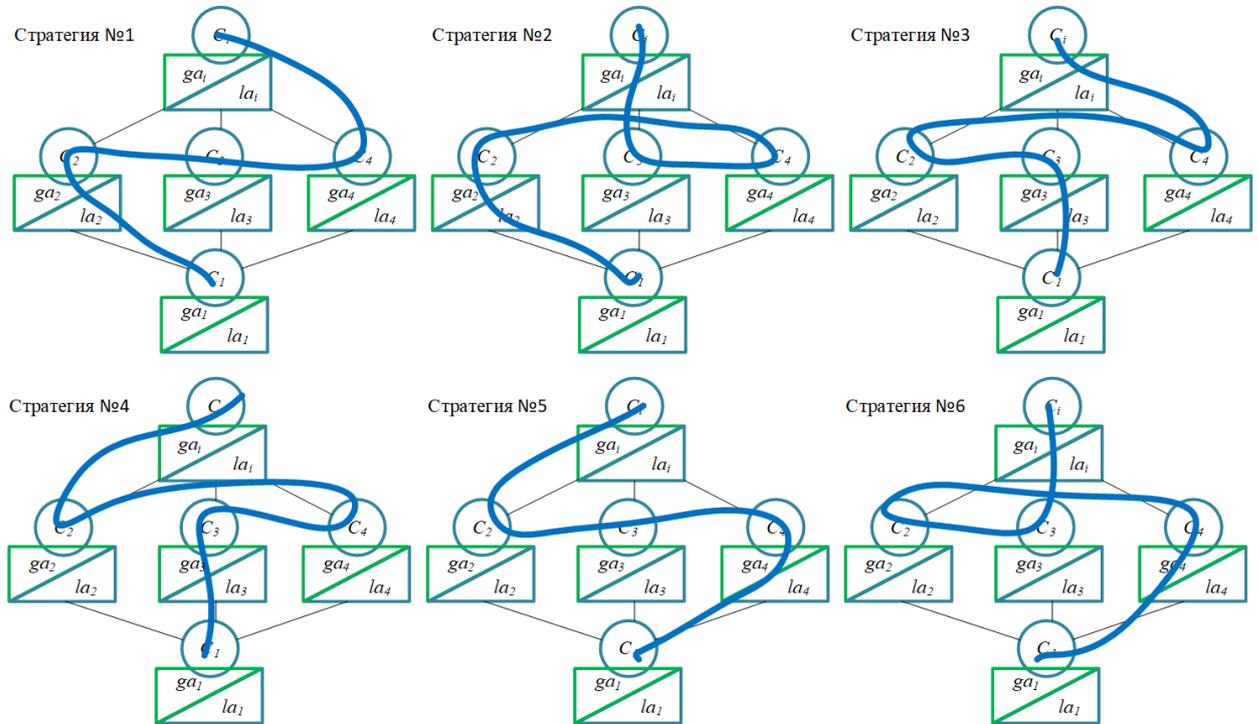


Рисунок 2.15 – Пример вариантов стратегий освоения пространства знаний в игровом контексте

В общем случае, каждому обучающему действию la_i в игровом сценарии может быть сопоставлено одно (рисунок 2.16) или более одного игрового действия gai , (Рисунок 2.17), что позволяет расширить игровой сценарий и повысить его игровую привлекательность без потери логических связей обучающего сценария.

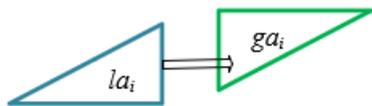


Рисунок 2.16 – Фрагмент пространства действий (сценарий «один-к-одному»)

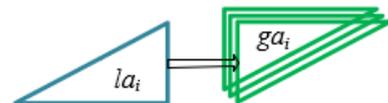


Рисунок 2.17 – Фрагмент пространства действий (сценарий «один-ко-многим»)

Таким образом, переносимая модель адаптации для адаптивных обучающих игр включает в себя модель предметной области в игровом контексте, представляемую структурно-упорядоченным пространством знаний; модель обучаемого, хранящую текущее состояние уровня знаний обучаемого на пространстве знаний; метод адаптации, основанный на свободном выборе обучаемым/игроком игровых действий, не нарушающим логику освоения пространства знаний, и контекстную модель, хранящую информацию о процессе взаимодействия обучаемого с игрой (рисунок 2.18).



Рисунок 2.18 – Модель адаптации

Предложенная модель применима для разработки игр различных жанров. Реализация модели в играх без сценария (например, sandbox games, quest games) заключается в разработке для каждого обучающего действия la_i , сопоставленного фрагменту c_i пространства знаний, его интерпретации в игровом контексте, при этом игровой процесс (gameplay) определяется только логикой процесса обучения. Разработка адаптивных обучающих игр с игровым сценарием дополнительно требует разработки способа интеграции процесса обучения с игровым процессом, сохраняющего логику как обучающего процесса, так и игрового сценариев.

2.4 Выводы по главе

В Главе 2 передана модель адаптации для адаптивных обучающих игр, применимая для разработки адаптивных обучающих игр различных жанров:

- разработана модель предметной области, представляемая пространством знаний, образуемым вложением множества фрагментов знаний, связанных логикой освоения, в решетку как алгебраическую структуру: каждому фрагменту знаний сопоставляется обучающее действие, требуемое для освоения этого фрагмента; каждому обучающему действию, выполненному обучаемым в процессе освоения пространства, сопоставляется оценка освоенности соответствующего фрагмента знаний, определяющая состояние обучаемого: при этом пространства действий и состояний структурно-эквивалентны пространству знаний;
- разработан метод интеграции процесса обучения с игровым процессом, основанный на сопоставлении каждому обучающему действию события игрового сценария как интерпретации обучающего действия в игровом контексте; при этом игровой сценарий разрабатывается как интерпретация обучающего курса, и действия обучаемого рассматриваются в контексте игрового процесса;
- разработана модель обучаемого/игрока, представляемая пространством состояний обучаемого/игрока, отображающим его текущий уровень знаний на пространстве знаний, интерпретированном в игровом контексте;
- метод адаптации процесса обучения к уровню знаний обучаемого основан на свободном формировании обучаемым стратегии по освоению пространства знаний в игровом контексте, определяемой текущим состоянием обучаемого и структурой самого пространства, обеспечивающий свободу выбора игровых действий, не противоречащих логике процесса обучения;

- для обеспечения совместимости, обучающего и игрового сценариев разработан метод динамического контентного согласования, основанный на генерации для любой последовательности независимых обучающих действий, выбираемых игроком в процессе игры, последовательности игровых действий, образующих связные фрагменты игрового сценария.

Глава 3. Автоматизация проектирования адаптивных обучающих игр

3.1 Процесс автоматизированной разработки адаптивных обучающих игр

На стадии разработки игры выделяют два ключевых этапа: проектирование и реализацию. Этап проектирования включает в себя разработку обучающего сценария, разработку игрового сценария, разработку модели адаптации, объединение обучающего и игрового сценариев и разработку архитектуры адаптивной обучающей игры. На стадии реализации выделяют такие процедуры как, программирование набора игровых решений, программирование модели адаптации, интеграцию модели адаптации с набором игровых решений и настройку механизмов адаптации.

Этапы и процедуры разработки адаптивной обучающей игры представлены на рисунке 3.1.

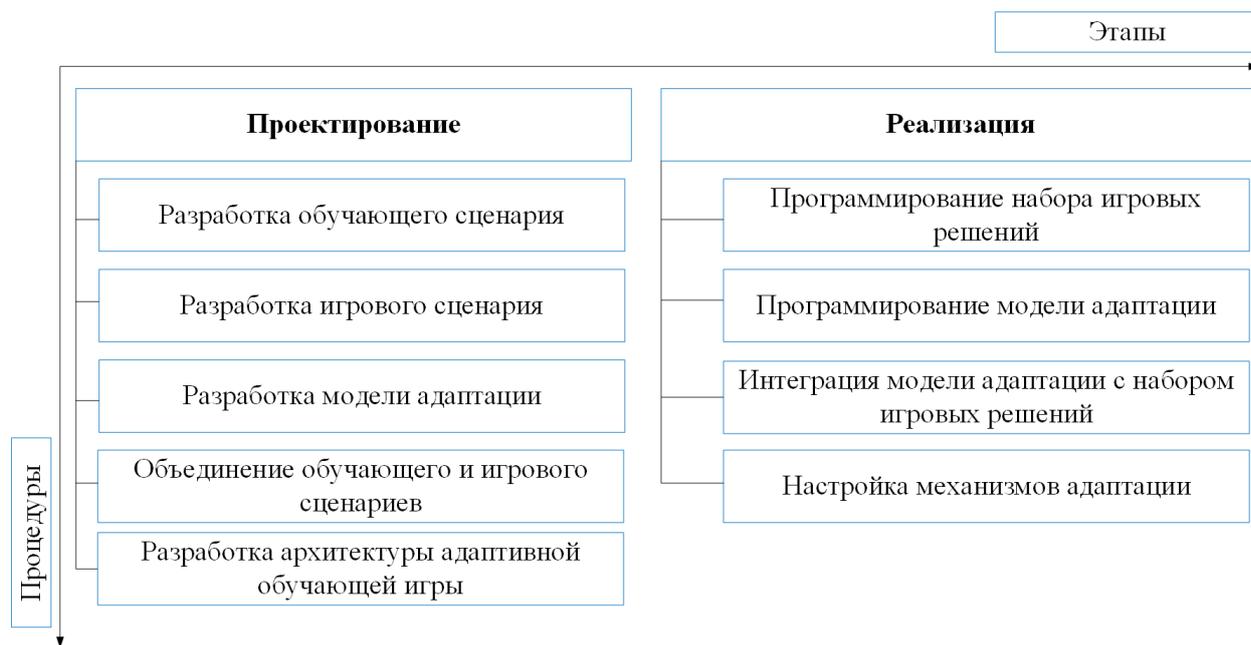


Рисунок 3.1 – Этапы и процедуры разработки адаптивной обучающей игры

Модель процесса разработки адаптивной обучающей игры представлена на рисунке 3.2.

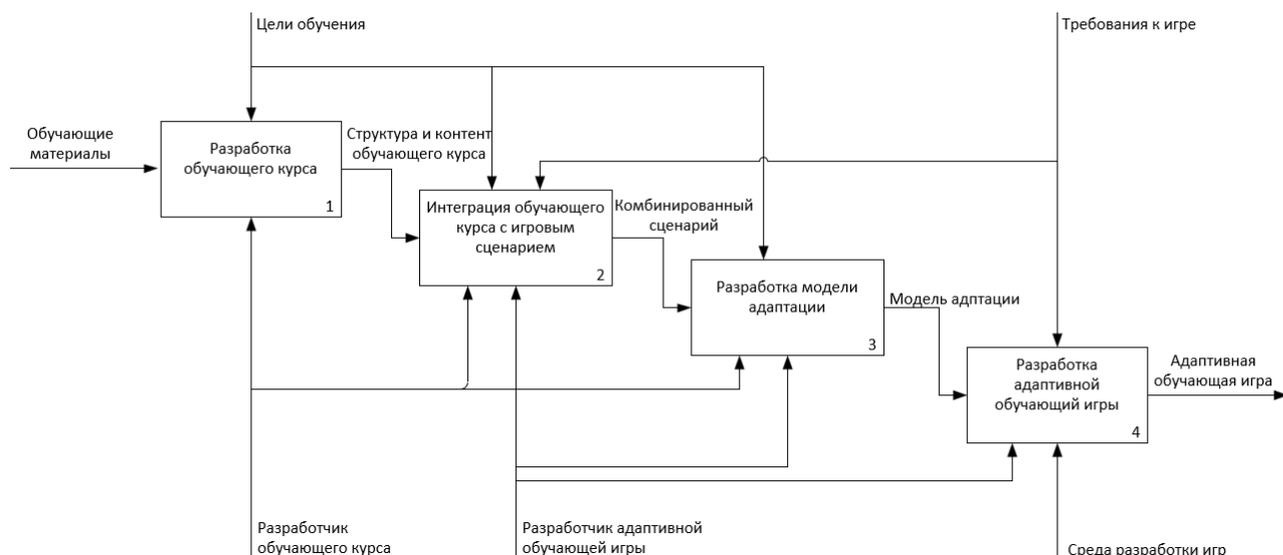


Рисунок 3.2 – Модель AS-IS процесса разработки адаптивной обучающей игры (нотация IDEF0)

Проведен анализ процесса разработки адаптивной обучающей игры и определены проектные и технологические процедуры автоматизированной разработки адаптивной обучающей игры.

На этапе проектирования, автоматизированы процедуры разработки модели адаптации и объединение обучающего и игрового сценариев, за счет использования переносимой модели адаптации. Типизирована процедура разработки архитектуры адаптивной обучающей игры, за счет применения готового шаблона.

На этапе реализации, автоматизирована процедура программирование модели адаптации, за счет использования готовых программных решений. Типизирована процедура интеграции модели адаптации, за счет использования алгоритма для интеграции модели адаптации.

Этапы и процедуры автоматизированной разработки игры представлены на рисунке 3.3.

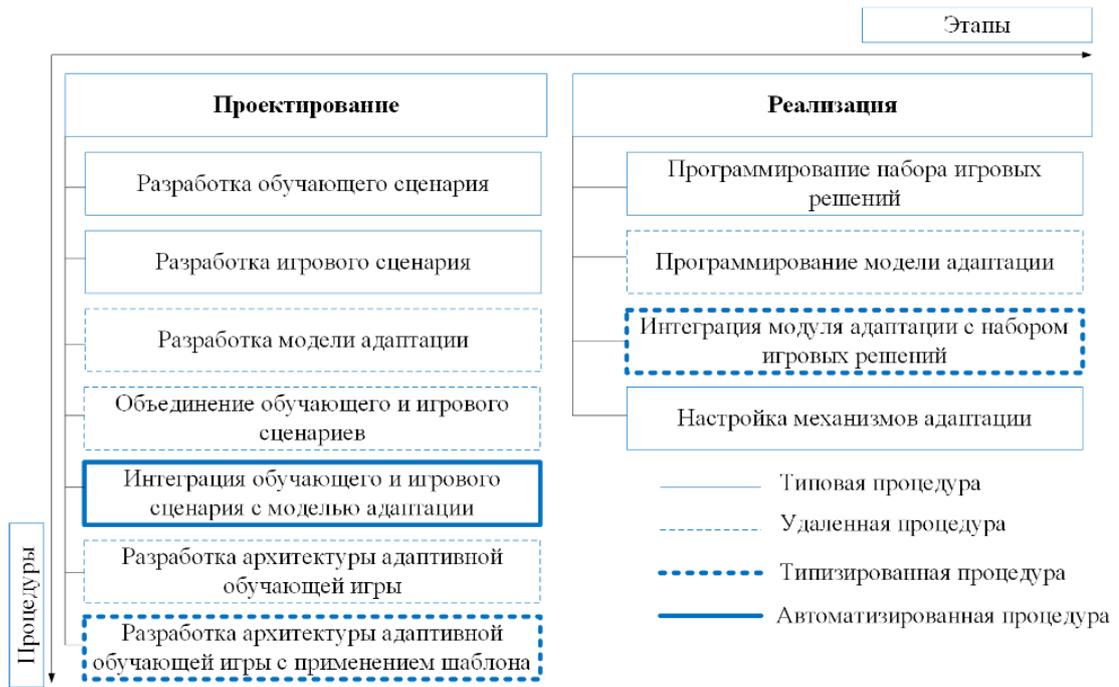


Рисунок 3.3 – Этапы и процедуры автоматизированной разработки адаптивной обучающей игры

Модель процесса автоматизированной разработки адаптивной обучающей игры представлена на рисунке 3.4.

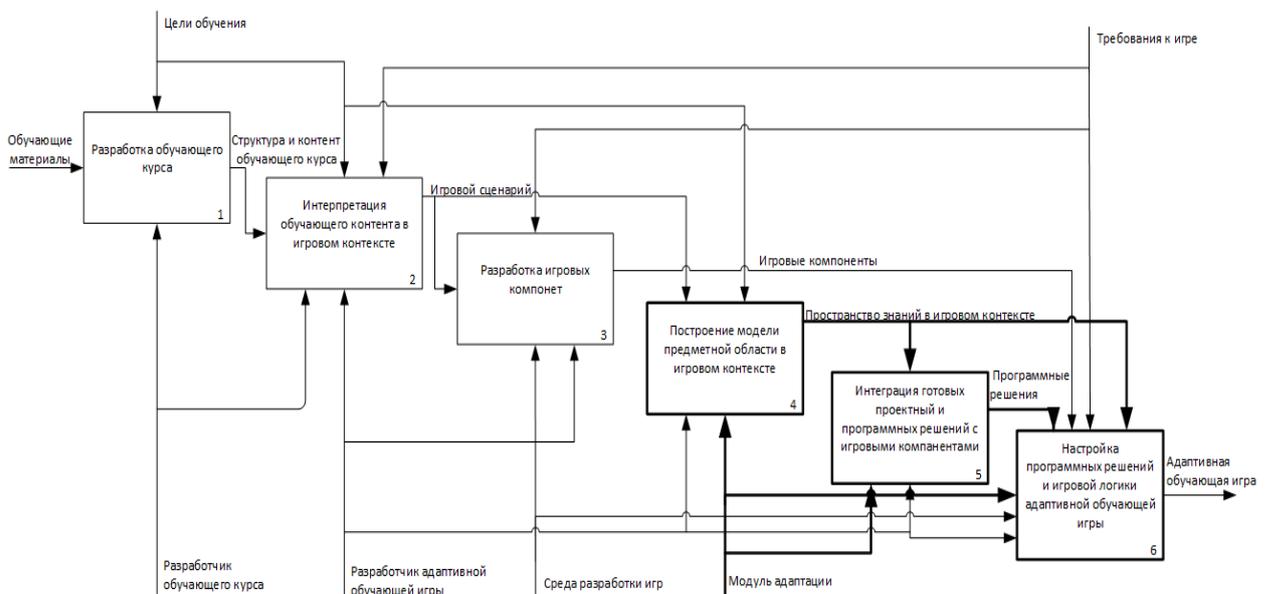


Рисунок 3.4 – Модель TO BE процесса автоматизированной разработки адаптивной обучающей игры (нотация IDEF0)

3.2 Шаблон архитектуры адаптивных обучающих игр

В контексте автоматизации проектирования адаптивных обучающих игр ключевой задачей становится создание универсального инструментария, обеспечивающего стандартизацию разработки данного типа игр. Для решения этих задач разработан шаблон архитектуры адаптивной обучающей игры (рисунок 3.5), включающий модуль адаптации и набор игровых решений.

Модуль адаптации включает в себя:

- модуль сбора данных, который фиксирует действия обучаемого (клики, перемещение, выбор ответа), игровые события (завершение заданий, активация триггеров, ошибки) и осуществляет предварительную обработку информации о взаимодействии пользователя с игрой, передавая данные в модель адаптации и БД;
- модель адаптации, которая осуществляет функцию анализа сырых данных из модуля сбора данных, сверяет полученную информацию с данными из БД и динамически настраивает сюжетные ветки в зависимости от выбора обучаемого. Таким образом полученные сырые данные преобразуются в персонализированные стратегии;
- БД, которая хранит данные об обучаемом и его действия в игре, игровые события, игровые и обучающие задания;
- система обратной связи, которая осуществляет передачу адаптивных корректировок (персонализированных стратегий, текущих диалогов заданий, игровых событий), замыкая цикл «анализ → решение → действие», обеспечивая непрерывную адаптацию.

Таким образом модуль адаптации отвечает за адаптивные корректировки в процессе взаимодействия обучаемого с игровым миром, анализируя данные о действиях пользователя и положениях пользователя в игровом мире, персонализируя сюжетные и обучающие стратегии в игре. Разработанный шаблон позволяет сократить время и ресурсы на разработку за счет переиспользуемых решений.



Рисунок 3.5 – Шаблон архитектуры адаптивной обучающей игры

3.3 Модуль адаптации

Модуль адаптации, основанный на предложенной модели адаптации, предназначен для функции построения пространства знаний и функции управления процессом обучения в игровом контексте. Разработано математическое обеспечение модуля адаптации, включающее алгоритмы построения пространства знаний, представляемого решеткой, алгоритм интеграции модуля адаптации в адаптивную обучающую игру. Модуль адаптации, включает инструмент для построения пространства знаний и набор решений для реализации метода адаптации в разрабатываемой адаптивной обучающей игре. Решения по выбору способов реализации разделены на независимые и зависимые от платформы, и интегрируются на этапе разработки игровых решений (рисунок 3.6).



Рисунок 3.6 - Архитектура модуля адаптации

3.4 Разработка платформозависимого набора решений модуля адаптации

Инструмент для построения пространства знаний реализован как Web приложение на языке Python, а визуализация пространства в среде Jupyter Notebook. Библиотеки, используемые для реализации данного приложения: tkinter, networkx, pyvis, numpy, matplotlib. Для визуализации пространства в среде jupyter notebook использована библиотека matplotlib, являющаяся надстройкой к библиотеке графов networkx.

Алгоритм построения пространства знаний представлен на рисунке 3.7



Рисунок 3.7 – Алгоритм построения пространства знаний

Исходными данными для построения пространства знаний, является структура обучающего курса в виде матрицы смежности соответствующего ориентированному графу. Процесс построения пространства знаний включает

проверку структурной-упорядоченности исходной структуры, и построения пространства.

Пример построения пространства с пояснением фрагментов кода, реализующий алгоритм построения пространства знаний представлен в приложении А.

3.5 Анализ игровых движков

В настоящее время для разработки игр применяют различные игровые движки: Unity3D, Unreal Engine, Rage Engine, Project Anarchy, App Game Kit, Game Maker: Studio. Которые различаются функциональными возможностями и сложностью разработки [95, 96].

При выборе игрового движка для разработки модуля необходимо опираться на требования к функционалу. Ключевыми характеристиками будем считать язык разработки, поддерживаемые платформы, лицензию распространения, наличие средств для разработки UI, ресурсоемкость (Таблица 3.3). Вторичными характеристиками будут служить сложность разработки игр в данной среде и комьюнити.

Таблица 3.3 – Анализ ключевых характеристик на применимость модели адаптации

Движок	Язык разработки	Поддерживаемые платформы	Лицензия распространения	Наличие средств для разработки UI
Unity3D	C#, JavaScript	PC, Xbox 360, Xbox One, Wii, PlayStation3, PlayStation4, PlayStation Vita, Windows Phone, iOS, Android, Native Client, Adobe Flash, Smart TV, WebGL	indie-версия: free software, pro-версия: 1500\$	+
Unreal Engine	C++	PC, Android, iOS, Windows Phone,	free software (MIT)	+
Cry Engine	C#	PC, PlayStation4, Xbox One	free software (для некоммерческого использования)	+

Продолжение таблицы 3.1

Движок	Язык разработки	Поддерживаемые платформы	Лицензия распространения	Наличие средств для разработки UI
Rage Engine	C#	PC, Xbox 360, Xbox One, PlayStation 3, PlayStation 4, Wii	proprietary software	+
Project Anarchy	Lua, C++	PC (только Windows), Android, iOS	free software	+
Game Salad	Drag and drop	PC, Android, iOS	free software	+
Game Maker: Studio	Gml	PC, Xbox One, PlayStation3, PlayStation4, PlayStation Vita, Windows Phone, iOS, Android, WebGL	free software	-
App Game Kit	AppGameKit Script	PC, Android, iOS, Blackberry, Raspberry	free software (оплачивается платформа)	-
Cocos 2D	C++	PC (MacOS), iOS	free software	+

Исходя из анализа данных характеристик выбор пал на движок Unity 3D. Причиной данного выбора послужило множество факторов: язык разработки C#, который проще C++, бесплатная лицензия, гибкий UI интерфейс, поддержка почти всех платформ и низкая ресурсоемкость. Важным фактором выбора данного движка стало огромное комьюнити, простота в разработке, большая библиотека бесплатных ассетов и постоянные обновления от разработчиков. Также в данный момент Unity 3D является самым популярным движком для разработки компьютерных игр.

3.6 Разработка платформозависимого набора решений

Платформозависимый набор решений представлен в виде готовых префабов и скриптов на Unity, а его структура представлена на рисунке 3.8.

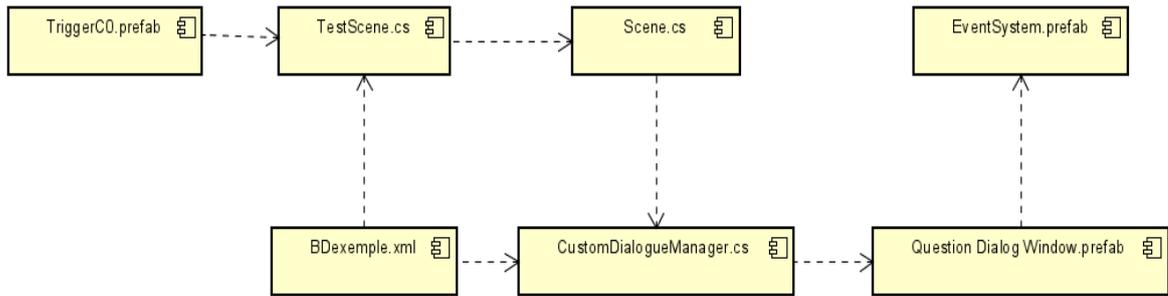


Рисунок 3.8 – Структура платформозависимого набора решений компоненты

3.6.1 Реализация метода триггеров

Для динамического контентного согласования, обучающего и игрового процессов реализован метод триггеров, позволяющий активировать систему диалоговых окон, определять положение игровых объектов в сцене и запускать игровые события, сопоставляемые обучающим действиям (рисунок 3.9). Триггеры представляются в виде игровых объектов, видимых и не видимых для пользователя в сцене.

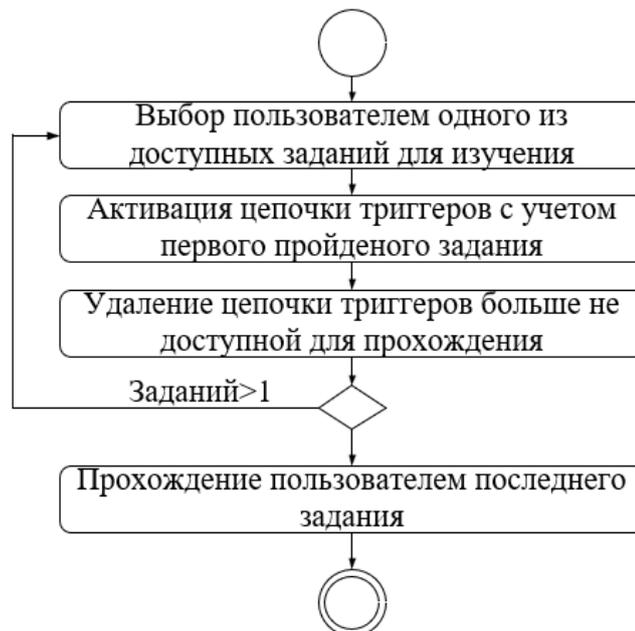


Рисунок 3.9 – Алгоритм управления процессом обучения в игровом контексте

Скрипт TestScene.cs выполняет роль триггера, при столкновении персонажа с данным объектом будет открываться система диалогов с указанным id элемента диалога.

Для создания нового триггера необходимо добавить в свойство любого объекта скрипт TestScene и свойство физического объекта RigidBody. Для столкновения с объектом он не обязательно должен быть видимым.

Для работы триггера необходимо заполнить его публичные свойства (рисунок 3.10):

1. Player – игровой персонаж или объект сцены при столкновении с которым будет открываться система диалогов.
2. Size – количество файлов БД.
3. File Name - имя файла БД.
4. Status – параметр который автоматически меняется в ходе выполнения заданий из указанного БД (1 – все задания из БД выполнены).
5. Start Question – id диалога, задания или события.

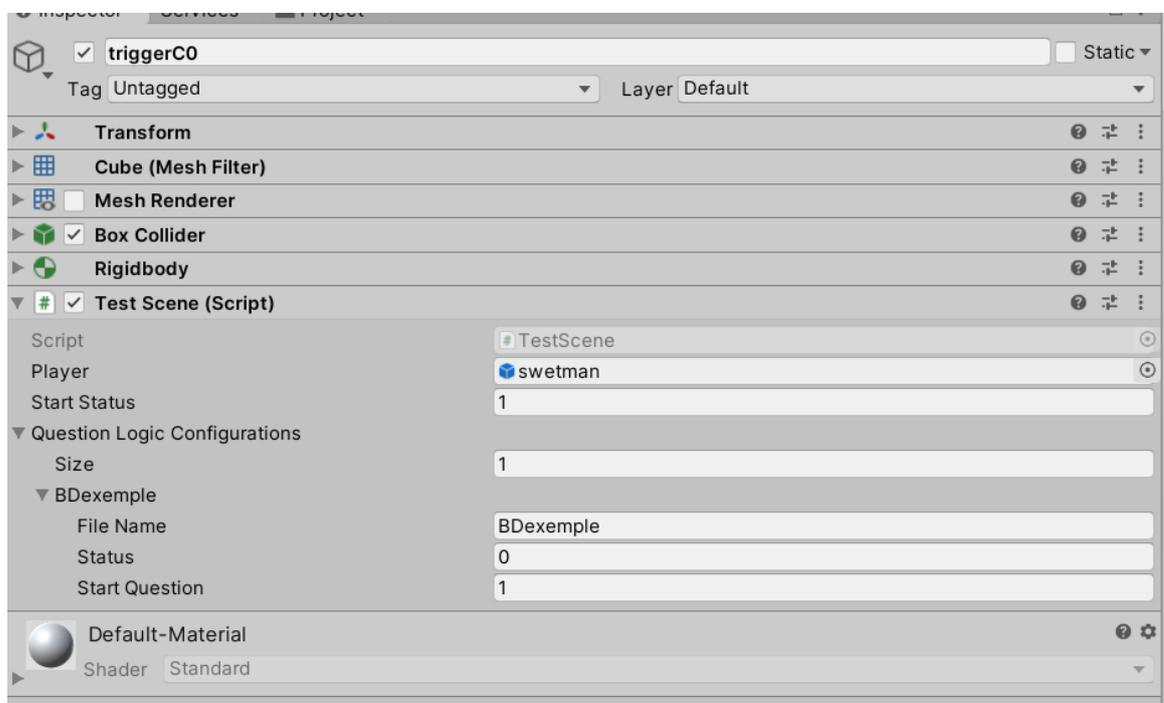


Рисунок 3.10 – Свойства триггера

3.6.2 Реализация системы диалоговых окон

Для организации диалоговых заданий в игре разработана система диалоговых окон, которая включает в себя шаблоны окон с вводом ответа задания, с выбором ответа задания, с выводом диалогов и выводом подсказки.

Система диалогов реализована на основе *Scroll View* методом кеширования элементов UI, чтобы упростить доступ к ним. Такая конструкция окна была выбрана для оптимизации работы системы диалога, вместо того чтобы каждый раз использовать создание/удаление объектов, можно просто обновлять те, которые уже есть.

Описание используемых элементов при отрисовке системы диалогов в зависимости от выбранного типа окон:

- диалоговое окно `message`: скролл, заголовок диалога, кнопку «ОК» (рисунок 3.11);
- диалоговое окно `messagetwo`: скролл, заголовок диалога, кнопку «ОК», кнопку «ВЫХОД», подсказку, информирование об ошибке, поле для ввода ответа (рисунок 3.12);
- диалоговое окно `messagethree`: скролл, заголовок диалога, кнопку «ВЫХОД», подсказку, кнопку варианта ответа, информирование об ошибке, поле для ввода ответа (рисунок 3.12).

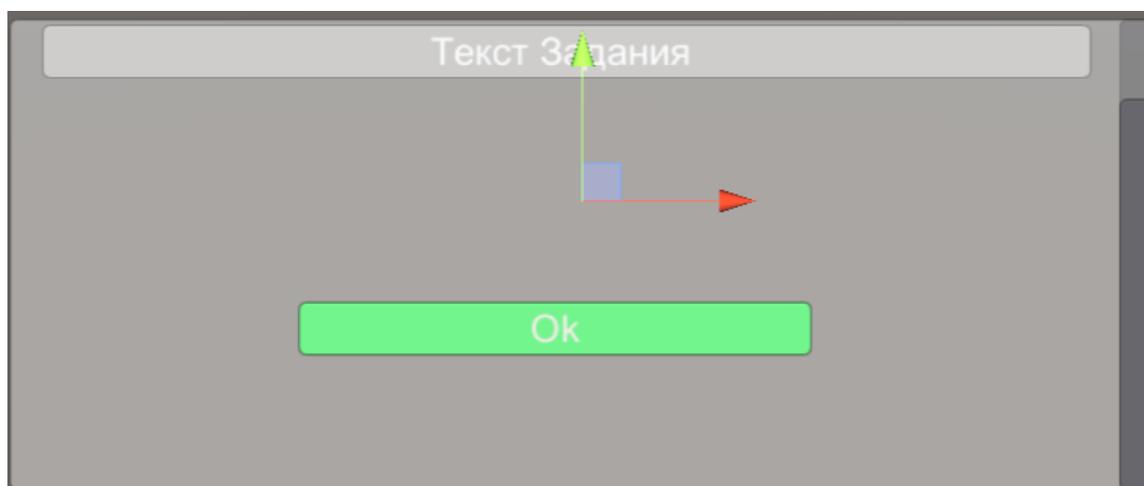


Рисунок 3.11 – Диалоговое окно `message`

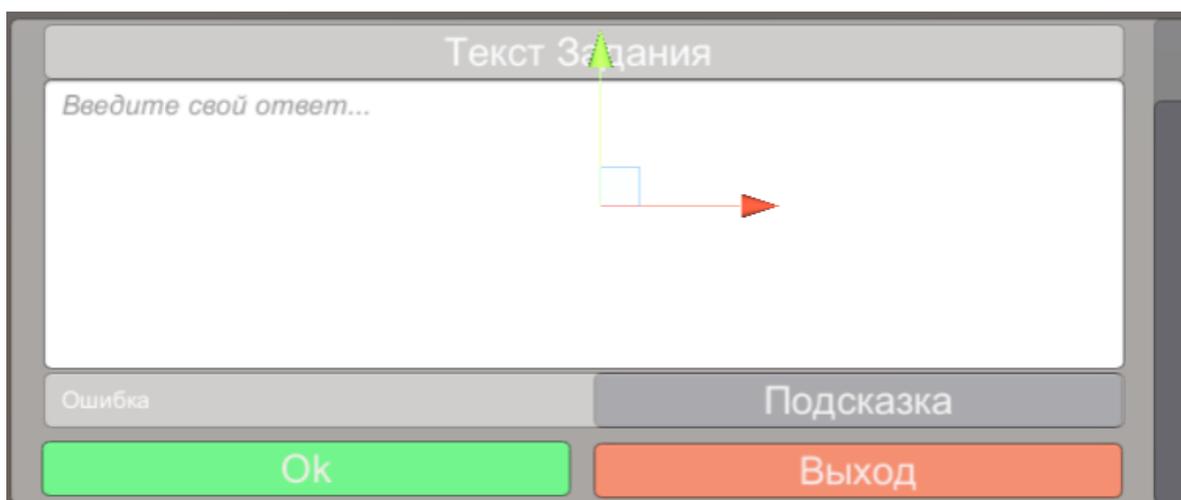


Рисунок 3.12 – Диалоговое окно messagetwo

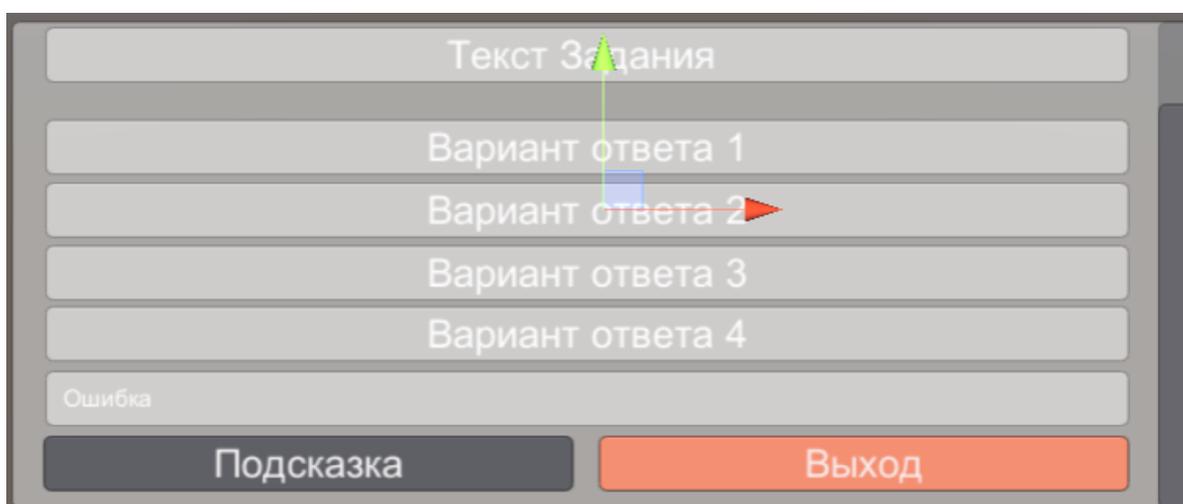


Рисунок 3.13 – Диалоговое окно messagethree

Скрипт *CustomDialogueManager.cs* отвечает за активацию окна диалога с заданием и переход на следующий диалог (или игровое событие) и сравнение с вариантами ответа, заполненными в БД. Позволяет задавать количество ошибок перед выводом подсказки (Рисунок 3.14).

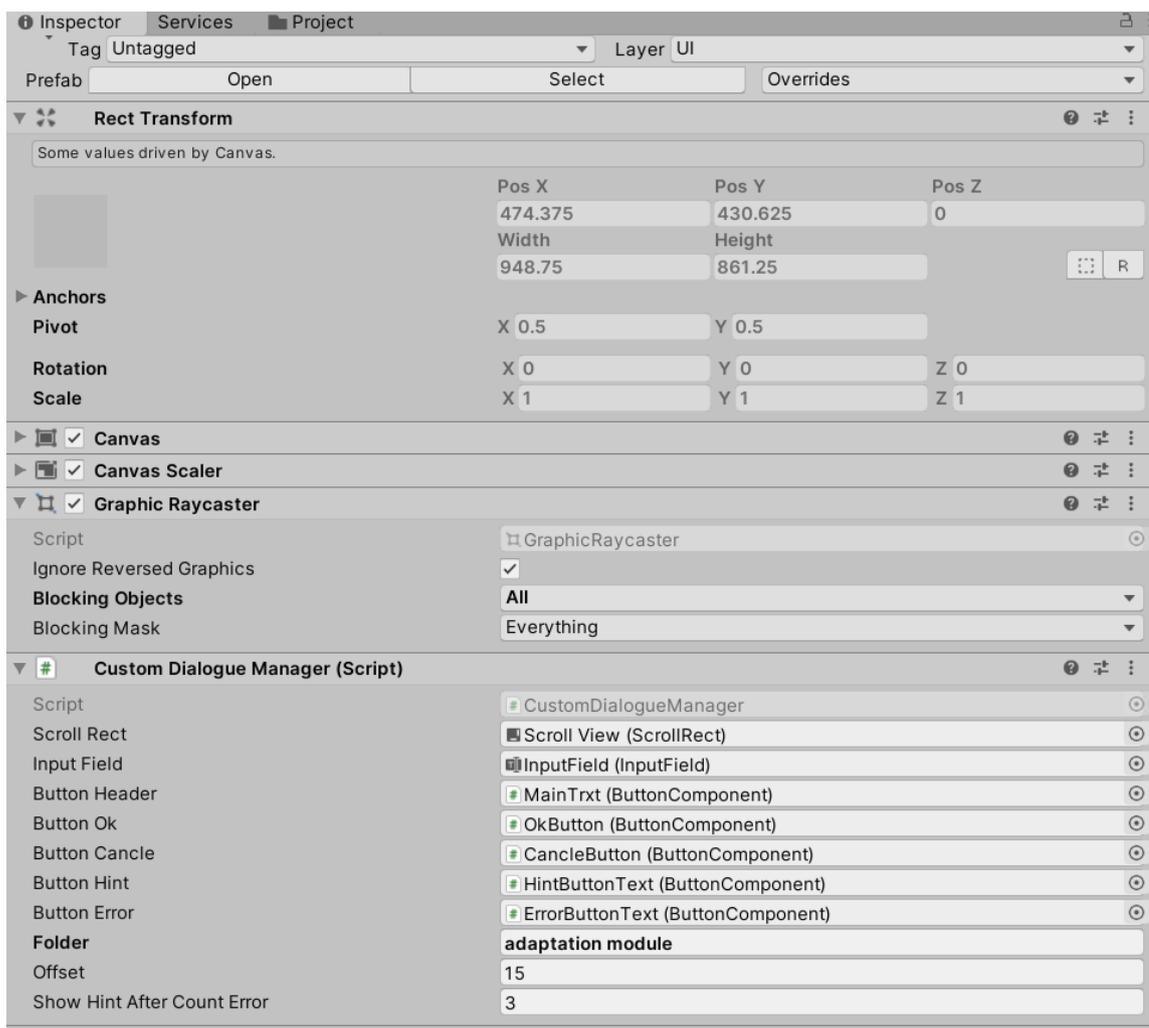


Рисунок 3.14 – Свойства системы диалогов

3.6.3 Разработка БД

Для хранения заданий, ответов на задания и игровых событий, сопоставленных обучающим заданиям и профиля игрока, разработана база данных на основе XML файла. Такой формат был выбран исходя из следующих преимуществ:

- само-описывающий (разметка описывает данные);
- легко обеспечить его обработку различным программным обеспечением, он хранит данные в виде древовидной структуры (визуально воспринимаемой человеком);
- легкая переносимость от одного приложения к другому.

Информационная структура БД представлена в таблице 3.2.

Таблице 3.2 – Информационная структура БД

Наименование	Тип	Комментарий
id	int	Идентификатор
string	text	Текст вопроса
string	dialogeName	Имя файла диалога, из которого в следующем вопросе брать информацию
string	status	Статус ответа
int	toQuestion	Следующий вопрос
bool	exit	Финальное задание
string	answers	Ответ
string	hint	Подсказка к вопросу
string	type	Тип вопроса: диалог (dialog, по умолчанию), информационное сообщение(message), custom
string	gameObjectName	Имя объекта, к которому привязан класс Scene
string	methodName	Имя вызываемого метода у объекта gameObjectName класса className
string	className	Имя класса, который принадлежит объекту gameObjectName

Разработанные технологические решения позволяют реализовывать четыре варианта игровых событий:

- диалог без задания,
- диалог с заданием,
- игровое событие и диалог с выбором ответов.

Структура заполнения БД представлена в таблице 3.3.

Таблице 3.2 – Структура заполнения БД

Название события	Структура заполнения
Диалог без задания	<code><question id="1" text="Текст диалога" toQuestion="2" type="message"> </question></code>
Диалог с заданием и окном ввода ответа	<code><question id="2" text="Текст задания" toQuestion="3" type="messagetwo"> <answer> <![CDATA[Ответ на задание]]> </answer> </question> <hint>Подсказка</hint></code>
Диалог с заданием и выбором ответа	<code><question id="2" text="Текст задания" toQuestion="3" type="messagethree"> <answer> <![CDATA[Ответ на задание]]> </answer> </question> <hint>Подсказка</hint></code>
Игровое событие	<code><question id ="3" type="custom" gameObjectName="Имя объекта, к которому привязан класс" className="TestScene" methodName="Название метода" /></code>

Для реализации игрового события необходимо создать метод в скрипте Scene.cs, который осуществляет проверку статуса диалоговых окон и хранит игровое событие соответствующие обучающему. При открытом диалоговом окне отключает функции анимации и движения персонажа.

3.7 Интеграция модуля адаптации с набором игровых решений

Модуль адаптации интегрируется с набором игровых решений на этапе реализации игры. Если адаптивная обучающая игра реализуется на движке Unity, то для ее разработки используется готовые платформозависимые решения. При выборе другого движка для реализации игры, разработчик опирается на предложенный пример и реализует модуль по аналогии с учетом специфики движка и его возможностей. Алгоритм интеграции модуля адаптации с набором игровых решений представлен на Рисунке 3.15.

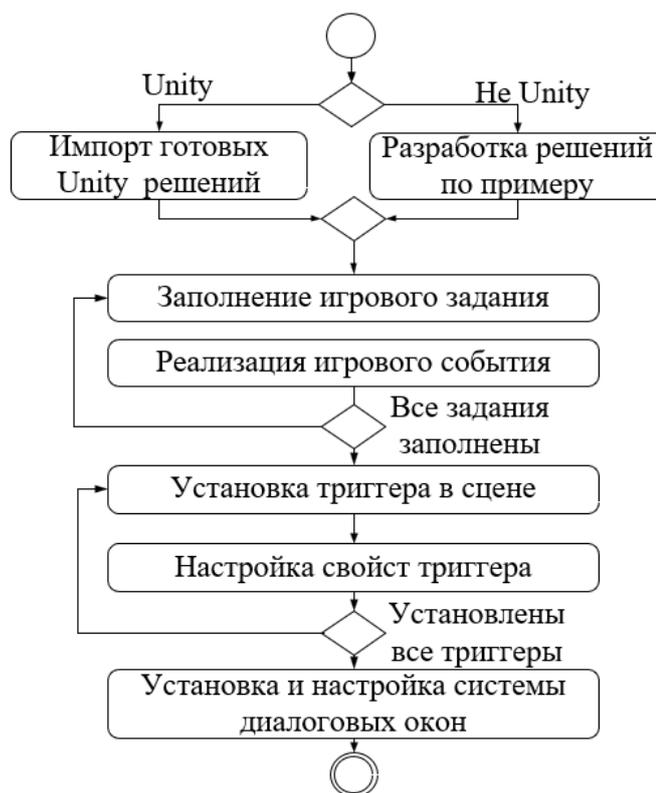


Рисунок 3.15 – Алгоритм интеграции модуля адаптации с набором игровых решений

3.8 Разработка метода взаимодействия проектировщиков обучающего контента и разработчиков адаптивной обучающей игры

Предложен метод взаимодействия проектировщиков обучающего контента и разработчиков адаптивной обучающей игры с модулем адаптации, реализующий единый подход к описанию обучающего и игрового контента (рисунок 3.16). Проектировщик обучающего контента описывает структуру обучающего курса в виде матрицы смежности соответствующего ориентированного графа. Для ввода матрицы смежности предусмотрены два способа: с помощью интерфейса, или посредством заполнения файла *.txt* по представленному образцу. Разработчик адаптивной обучающей игры использует файл описания пространства знаний, построенного в модуле адаптации, и фрагментов игрового для заполнения шаблона БД и привязки игровых событий к обучающему сценарию.



Рисунок 3.16 – Взаимодействие участников процесса разработки адаптивной обучающей игры

3.9 Метод автоматизированного проектирования адаптивных обучающих игр

Для разработки адаптивных обучающих игр на основе предложенной модели адаптации разработан метод автоматизированного проектирования адаптивных обучающих игр, основанный на шаблонной архитектуре, включающий набор платформозависимых и платформонезависимых компонент, интегрируемых с игровыми решениями, реализующий алгоритмы построения игрового сценария, управления процессом обучения в игровом контексте и интеграции модели адаптации с набором игровых решений.

Метод автоматизированного проектирования адаптивных обучающих игр включает следующие проектные и технологические процедуры:

- описание обучающего контента адаптивной обучающей игры в виде матрицы смежности;

- описание фрагментов обучающего контента в игровом контексте;
- построение пространства знаний;
- разработка архитектуры адаптивной обучающей игры с применением шаблона;
- заполнение БД игрового сценария.

3.10 Выводы по главе

В Главе 3 описаны способы формализации и типизации проектных и технологических процедур, их реализация в технологии автоматизированной разработки адаптивных обучающих игр, и программный модуль, реализующий предложенную модель адаптации:

- определены проектные и технологические процедуры автоматизированной разработки адаптивной обучающей игры;
- разработан шаблон архитектуры адаптивной обучающей игры, включающий модуль адаптации, реализующий разработанную модель адаптации, и набор игровых компонент;
- спроектирован модуль адаптации, включающий инструмент для построения пространства знаний и набор платформозависимых и платформонезависимых решений для реализации метода адаптации в разрабатываемой адаптивной обучающей игре;
- разработаны алгоритмы, реализующие проектные и технологические процедуры автоматизированной разработки адаптивной обучающей игры;
- предложен метод взаимодействия проектировщиков обучающего контента и разработчиков адаптивной обучающей игры с модулем адаптации, реализующий единый подход к описанию обучающего и игрового контента.

Глава 4. Апробация моделей и методов на разработке адаптивной обучающей игры

4.1 Прототип обучающей игры

В качестве прототипа была взята ролевая обучающая игра «Камми» для изучения объектно-ориентированного программирования (ООП), реализованная в жанре *escaperoom* [84]. По сюжету игры главным персонажем является «Профессор Камаев» (с разрешения профессора кафедры САПР и ПК Волгоградского государственного технического университета, д.т.н., Камаева Валерия Анатольевича), сознание которого в результате неудачных опытов (рисунок 4.1) переместилось в маленького робота, состоящего из 16 магнитов – Камми. В своем новом состоянии профессор оказывается в мире больших вещей, о котором он ничего не знает и в котором он не умеет жить. Для того чтобы вернуть профессору его человеческий облик, игрок должен провести Камми до лаборатории и починить установку. Для этого ему предстоит освоиться в новом для него мире и выполнить множество игровых заданий. В процессе игры игрок постепенно расширяет игровой мир, добавляя новых персонажей, принимает новые формы, обучается новым возможностям. Для этого он разрабатывает новые классы с различными свойствами, и создает игровые персонажи, как объекты этих классов. Описание игрового мира, которое создает игрок, в контексте ООП представляется диаграммой классов его сущностей.

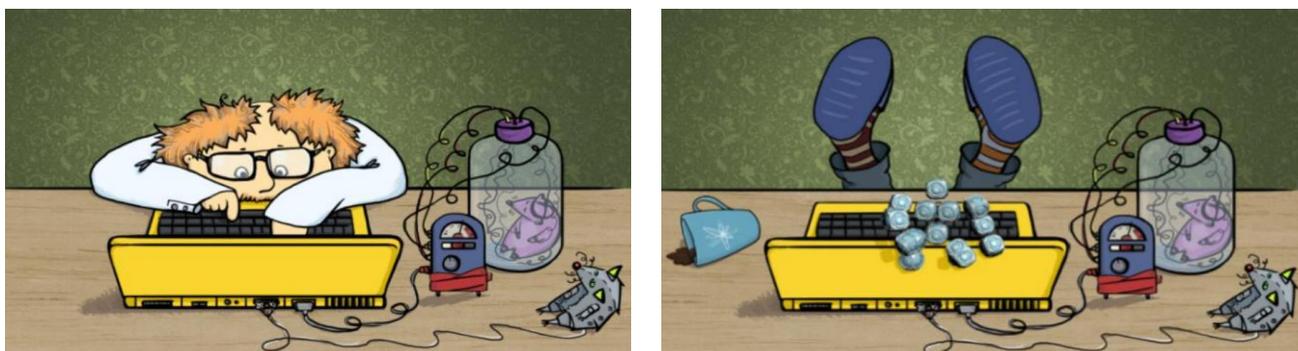


Рисунок 4.1 – «Профессор Камаев» в двух состояниях

Обучающий курс по ООП, реализованный в игре, включает множество разделов, интерпретированных в игровом контексте (таблица 4.1).

Таблица 4.1 – Разделы курса (фрагмент)

Идентификатор	Раздел курса
C_0	Типы данных
C_1	Разработка методов класса
C_2	Функции
C_3	Функции с параметром
C_4	Массивы

Исходная структура обучающего курса представляет собой граф, задающий конечное множество фрагментов знаний, составляющих содержание области знаний, и связи между фрагментами, отражающие логику освоения разделов обучающего курса.

В прототипе игры «Камми» структура курса, построенная разработчиком курса, свернута в линейную цепочку последовательно изучаемых разделов (рисунок 4.2). Соответственно игровой сценарий, сопоставленный обучающему сценарию, также является линейным.

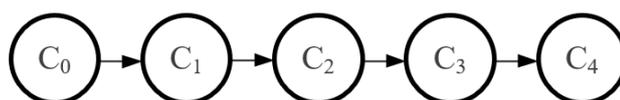


Рисунок 4.2 – Фрагмент структуры обучающего курса

4.2 Ad-hoc разработка адаптивной обучающей игры

В процессе анализа прототипа игры «Камми», у нее были выявлены следующие недостатки:

- плохо проработаны анимации;
- устаревший внешний вид интерфейса;
- низкое качество отрисовки игрового мира;

- наличие критических ошибок работы игры.

Большинство из выявленных недостатков обусловлены ограниченными возможностями движка, поэтому для их устранения их было решено портировать игру на более современный движок Unity.

4.2.1 Разработка набора игровых решений

В разработанной версии игры реализован игровой мир объекта «Кухня». При построении игрового мира были использованы стандартные 3D примитивы Unity (таблица 4.2) и готовые бесплатные 3D модели предметов интерьера кухни.

Таблица 4.2 – Соответствие 3D примитивов и объектов игрового мира

Модель игрового мира	Примитив Unity
Пол	Terrain
Препятствие	Cube и cylinder
Стена	Plane

Внешний вид сцены «Кухня» показан на Рисунке 4.3.

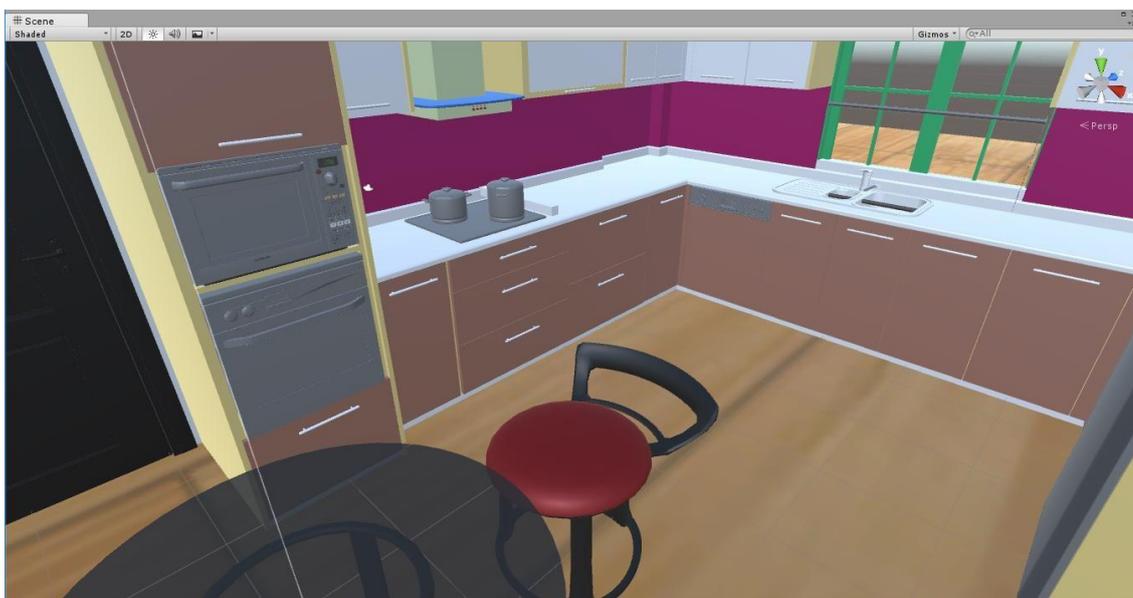


Рисунок 4.3 – Экранная форма игрового мира в среде Unity

Разработаны следующие модели персонажа: камми (рисунок 4.4), гиря

(рисунок 4.5), змея (рисунок 4.6), шар (рисунок 4.7). Каждая модель реализует соответствующую форму. Все 3D модели персонажа реализованы как различные комбинации из 16-ти отдельных шарообразных объектов, имитирующих магниты. Каждый объект представляет собой стандартный 3D примитив sphere с заданной текстурой (см. Рисунок 35). Размеры всех объектов одинаковы, за исключением одного объекта, который имеет гораздо больший размер и имитирует «голову» всех моделей.

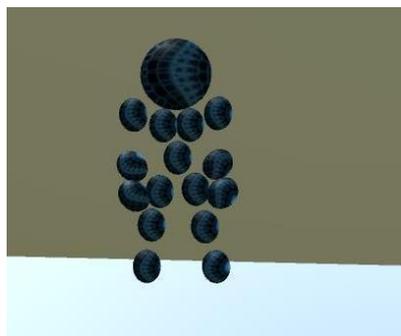


Рисунок 4.4 – Форма камни

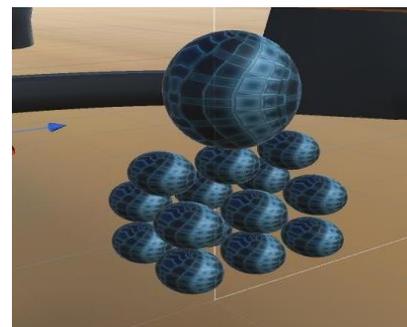


Рисунок 4.5 – Форма гиря

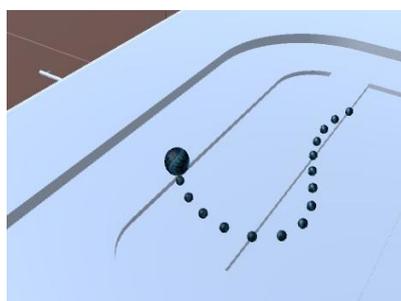


Рисунок 4.6 – Форма змея

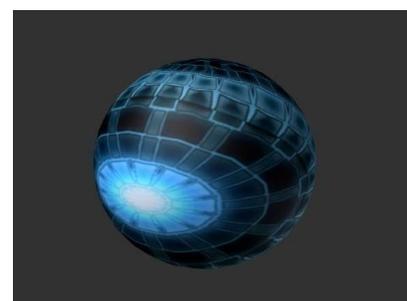


Рисунок 4.7 – Форма шар

Реализованы базовые функции для всех форм:

- форма камни: движение, движение с ускорением, прыжок, приседание, распадение;
- форма гиря: вертикальный удар;
- форма шар: качение, качение с ускорением;
- форма змея: ползание.

Реализованы анимации из Assets, которые отображены на дереве состояний (рисунок 4.8).

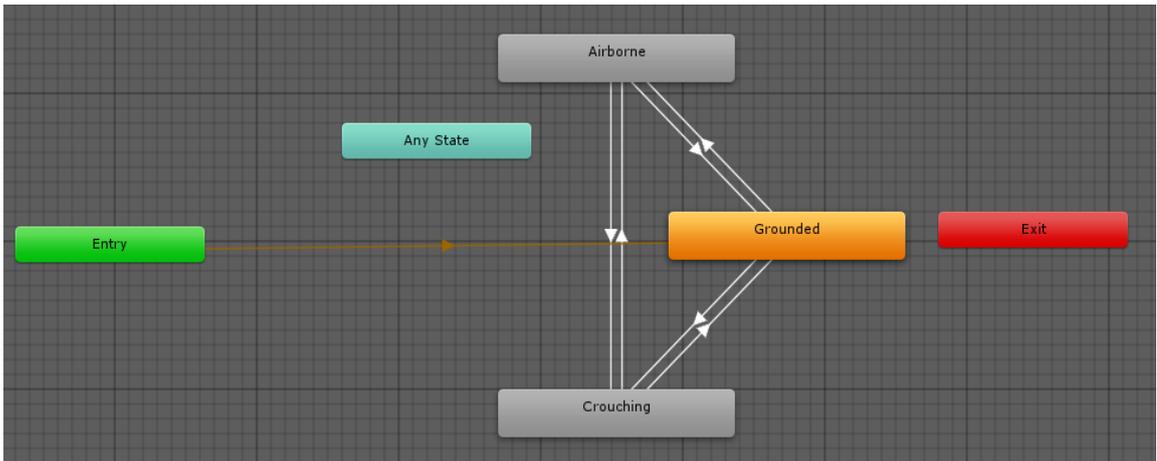


Рисунок 4.8 – Дерево состояний

В аниматоре (рисунок 4.9) было проработано дерево состояний, описывающее движения персонажа (ходьба, бег, повороты).

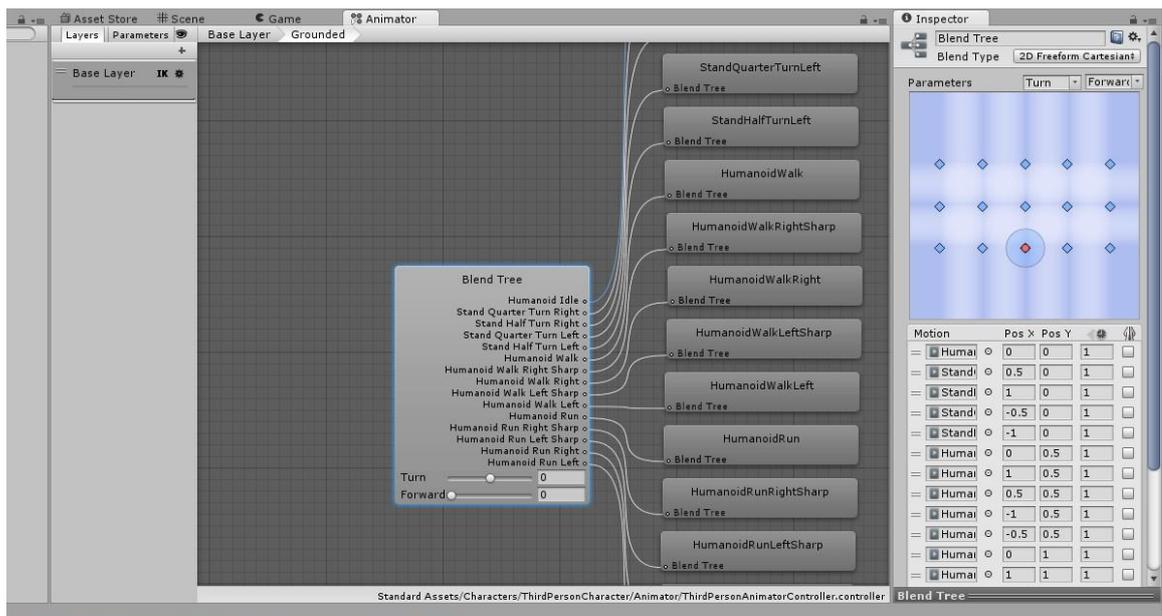


Рисунок 4.9 – Работа с аниматором в среде Unity

Настроены основные механики игры и реализованы игровые события в соответствии с игровым сценарием.

В среде Unity для настройки физики игры используется специальный компонент Rigidbody (рисунок 4.10). Данный компонент был добавлен на все формы персонажа с одинаковыми настройками: mass=1 (вес в килограммах),

drag = 0(вязкость среды), angulardrag=0, usegravity = true (гравитация), iskinematic=false.

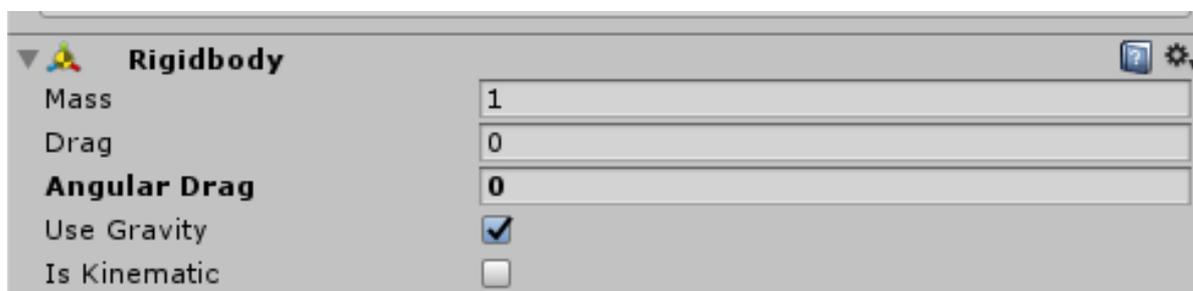


Рисунок 4.10 – Применение Rigidbody в среде Unity

Для того чтобы объекты могли сталкиваться и не проходили сквозь стены ко всем объектам применен Collider (рисунок 4.11).

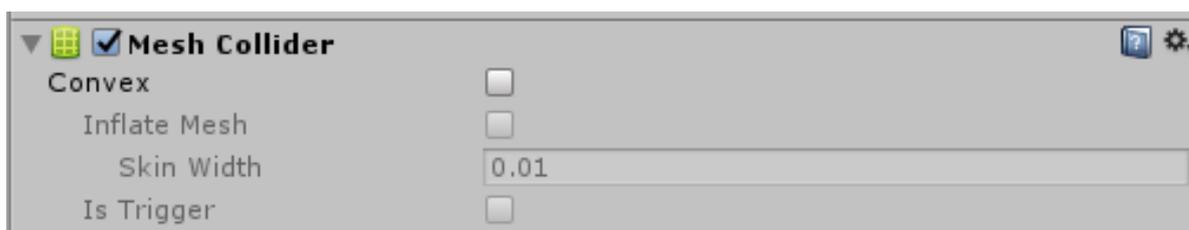


Рисунок 4.11 – Применение Collider в среде Unity

Настроены игровые камеры. В игровом мире располагаются четыре камеры (для каждой из форм персонажа по одной). В зависимости от активной формы персонажа используется соответствующая камера. Для управления камерой использованы стандартные скрипты *FreeLookCamuHandHeldCam*.

4.2.2 Разработка нелинейной структуры обучающего курса

Обучающий курс представляется набором связанных элементов. Каждому элементу структуры сопоставляется фрагмент обучающего курса. В качестве элементов курса могут выступать блоки теоретического материала (лекции, и/или их фрагменты), практические задания, тесты различного вида, упражнения и т.д.

Связи между элементами задают логическую связность курса, отражающую его внутреннее устройство.

Курс объектно-ориентированное программирование – это методология программирования, которая основана на представлении программы в виде совокупности объектов, каждый из которых является реализацией определенного класса (типа особого вида), а классы образуют иерархию на принципах наследуемости. Таким образом объектно-ориентированное программирование — это методика, которая концентрирует основное внимание программиста на связях между объектами, а не на деталях их реализации [97].

Выделяются два уровня, обучающего курса: базовый уровень и профессиональный. В контексте данной работе, на базовом уровне игрок изучает основные принципы ООП, разделы курса представлены в таблице 1. Профессиональный представляет собой режим свободной игры, где игрок расширяет игровой мир, и более углубленно развивает навыки программирования.

Произведена интерпретация разделов курса и тем обучающих заданий (таблица 4.3).

Таблица 4.3 – Интерпретация разделов курса и тем обучающих заданий

Раздел курса	Тема обучающего задания	Идентификатор
Типы данных	Выбор типа данных	C_0
Объекты и классы	Создание класса и объекта класса	C_1
Функции	Разработка метода движения вперед	C_2
	Разработка метода движения назад	C_3
	Разработка метода движения вправо	C_4
	Разработка метода движения влево	C_5
Функции с параметром	Разработка метода ускоренного движения вперед	C_6
	Разработка метода ускоренного движения назад	C_7
	Разработка метода ускоренного движения вправо	C_8
	Разработка метода ускоренного движения влево	C_9
	Разработка метода прыжка	C_{10}

Продолжение таблицы 4.3

Раздел курса	Тема обучающего задания	Идентификатор
Массивы	Заполнение одномерного массива формы Змея	C_{11}
	Заполнение двумерного массива формы Камми	C_{12}
	Заполнение трехмерного массива формы Гирия	C_{13}
	Заполнение одномерного массива формы Шара	C_{14}

Разработана нелинейная структура курса. Нелинейная структура курса в соответствии с разделами курса и темами обучающих заданий представлена на рисунке 4.12.

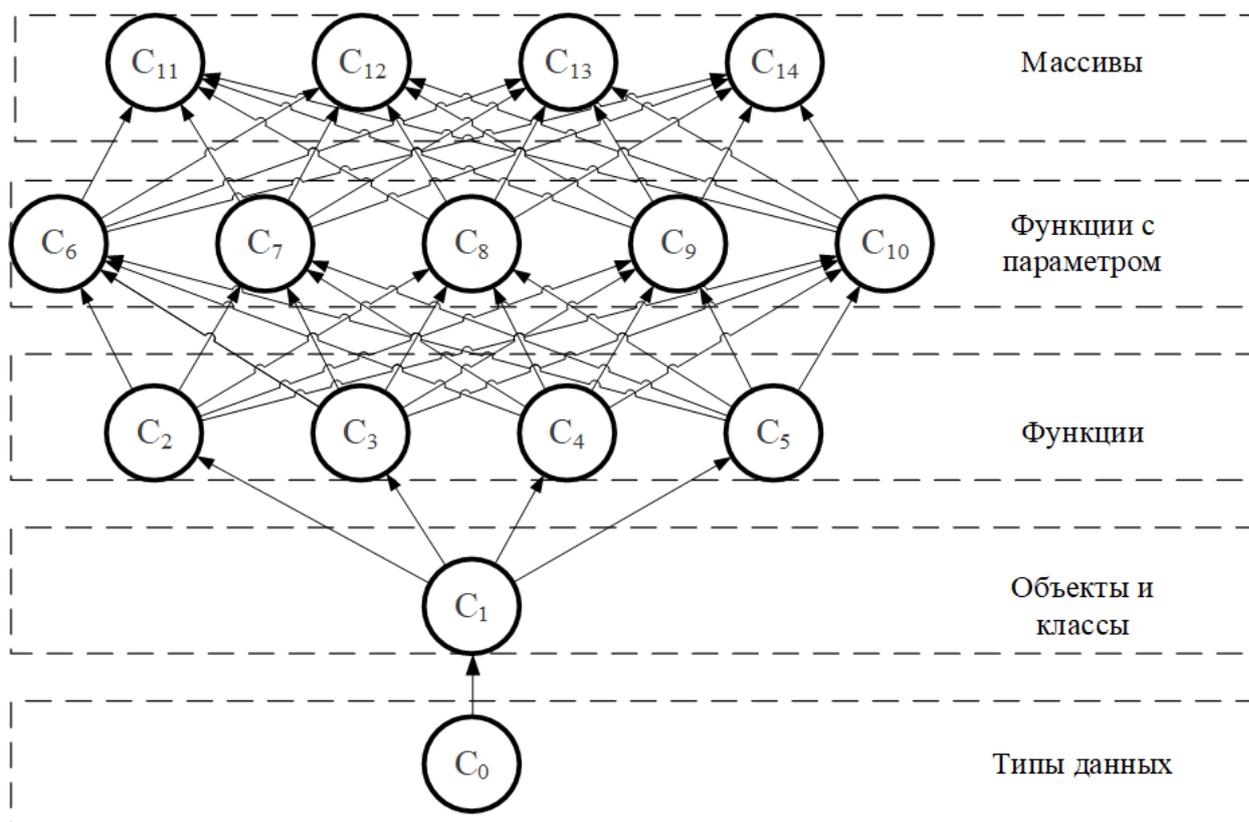


Рисунок 4.12 – Нелинейная структура курса

Комбинированный сценарий, который подразумевает слияние игрового и обучающего сценариев и имеет аналогичную структуру.

4.2.3 Построение пространства знаний

Построено пространство знаний без использования модуля для построения пространства знаний (рисунок 4.13).

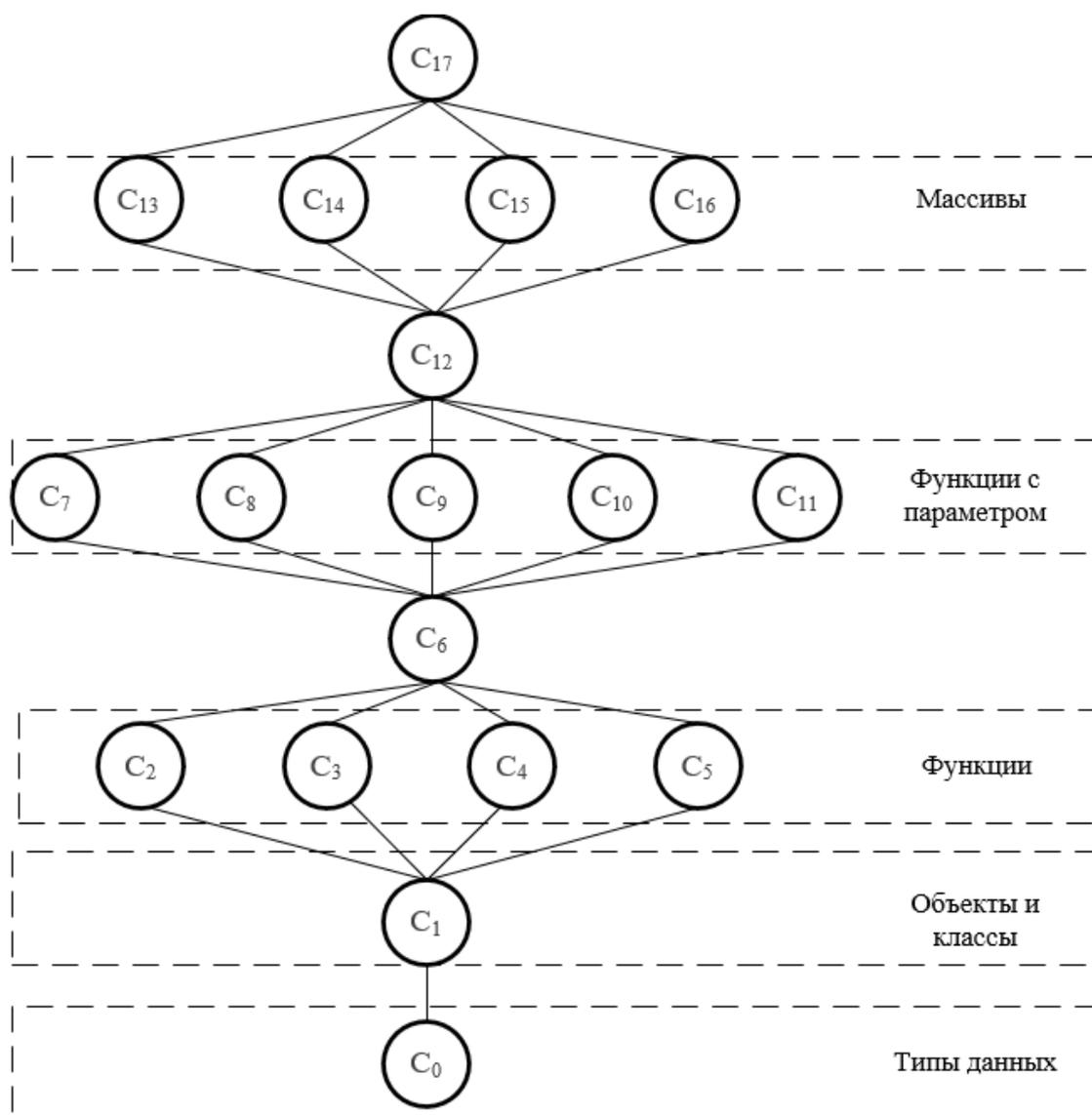


Рисунок 4.13 – Пространство знаний

При построении пространства знаний появляются добавочные элементы C_6 , C_{12} , C_{17} . Данные добавочные элементы не имеют определенного обучающего задания, а используются как промежуточные между разделами или завершающие задания. Описание информационных, обучающих или игровых заданий в них закладывается на этапе интерпретации обучающих действий в игровом контексте.

4.2.4 Интерпретация обучающих действий в игровом контексте

В игре «Камми» виртуальный мир интерпретирует объектно-ориентированную парадигму как систему правил жизни в виртуальном мире. Интерпретации разделов обучающего курса в игровом контексте представлены в таблице 4.4.

Таблица 4.4 – Интерпретации разделов обучающего курса в игровом контексте

Раздел курса	Интерпретация в игровом контексте	Реализация
Типы данных	Идентификация пользователя	Для хранения персональных данных игрок должен определить типы этих данных
Классы и объекты	Создание персонажа игры	Для заселения мира персонажами игрок должен описывать класс и создавать объекты этих классов
Функции	Управление персонажем игры	Для описания поведения персонажей игрок должен создавать и модифицировать функции (методы классов)
Массивы	Изменение формы персонажа игры	Для изменения формы персонажей игрок должен заполнять массивы

Для обеспечения одновременного достижения игровой цели и цели обучения игровые и обучающие задания выполняются одновременно, т.е. были объединены в одно действие.

Интерпретация обучающих заданий в игровом контексте показана в таблице 4.5.

Таблица 4.5 - Интерпретации обучающих заданий в игровом контексте

Идентификатор	Игровая интерпретация	Обучающее задание
C_0	Идентификация пользователя	Выбор типа данных
C_1	Создание персонажа игры	Создание класса и объекта класса
C_2	Обучение персонажа движению вперед	Разработка метода движения вперед

Продолжение таблицы 4.5

Идентификатор	Игровая интерпретация	Обучающее задание
C_3	Обучение персонажа движению назад	Разработка метода движения назад
C_4	Обучение персонажа движению вправо	Разработка метода движения вправо
C_5	Обучение персонажа движению влево	Разработка метода движения влево
C_6	Промежуточный диалог после выполнения обучаемым всех базовых заданий по управление персонажем игры	
C_7	Обучение персонажа ускоренному движению	Разработка метода ускоренного движения
C_8	Обучение персонажа движению на корточках	Разработка метода движению на корточках
C_9	Обучение персонажа перерезанию через препятствие	Разработка метода перерезания через препятствие
C_{10}	Обучение персонажа удару рукой	Разработка метода удара рукой
C_{11}	Обучение персонажа игры прыжку	Разработка метода прыжка
C_{12}	Промежуточный диалог после выполнения обучаемым всех заданий по управление персонажем игры	
C_{13}	Обучение персонажа принимать форму Змея	Заполнение одномерного массива формы Змея
C_{14}	Обучение персонажа принимать основную форму	Заполнение двумерного массива формы Камми
C_{15}	Обучение персонажа принимать форму Гиря	Заполнение трехмерного массива формы Гиря
C_{16}	Обучение персонажа принимать форму Шара	Заполнение одномерного массива формы Шара
C_{17}	Промежуточный диалог с информацией о прохождении базового уровня игры	

4.2.5 Реализация модели адаптации и настройка механизмов адаптации

В среде Unity предусмотрены инструменты UI для создания интерфейса игр, которые дают возможность создания структуры диалогового окна и окна редактирования программного кода.

Для разработки интерфейса диалогового окна (рисунок 4.14) были использованы следующие инструменты:

- «Canvas»;
- «Panel»;
- «Button»;
- «Text»;
- «Scroll View».

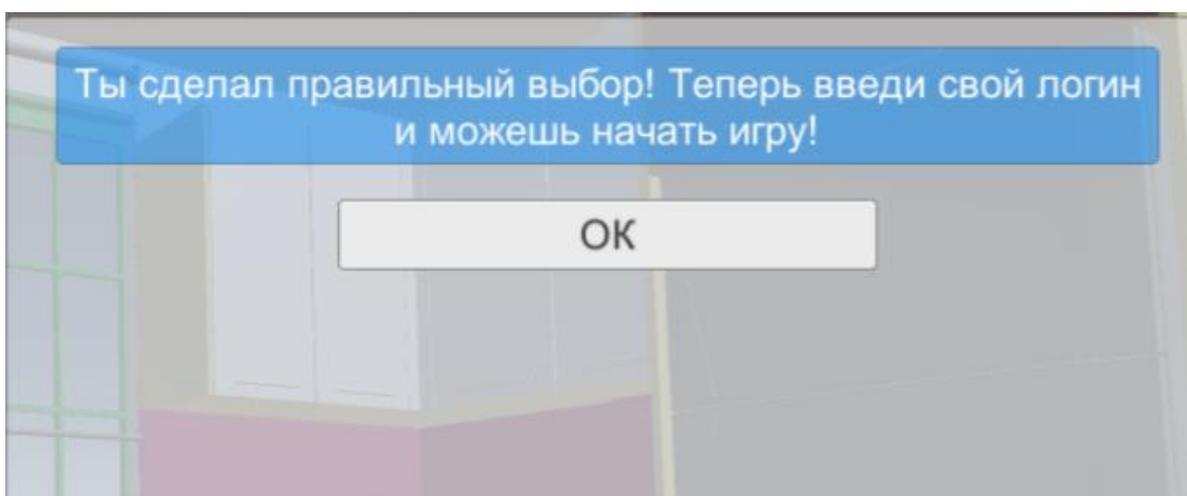


Рисунок 4.15 – Пример диалогового окна

Верификации программного кода заключается в сравнении ответа пользователя (в виде программного кода) на поставленный вопрос, с БД готовых ответов.

Разработана БД на основе *json* файла. Структура БД представлена в таблице 4.6.

Таблице 4.6 - Информационная структура БД

Наименование	Тип	Комментарий
id	int	Уникальный идентификатор вопроса
question_text	text	Текст вопроса
dialogeName	varchar(255)	Имя файла диалога, откуда берется следующий вопрос
status	enum	Статус ответа: correct, incorrect, pending

Продолжение таблицы 4.6

Наименование	Тип	Комментарий
toQuestion	int	Ссылка на id следующего вопроса
exit	boolean	Флаг финального вопроса (true/false)
answers	json	Массив правильных ответов в формате JSON

Пример заполнения диалогов, заданий, ответов в БД представлен на рисунке 4.16.

```

{
  "questions": [
    {
      "id": 1,
      "question_text": "Теперь нужно разработать алгоритм функции MoveForward. Положение любого объекта на плоскости определяется двумя целочисленными координатами",
      "answers": [ " y=y+1;" ],
      "hint": "y++; y=y+1;",
      "toQuestion": 2
    }
  ]
}

```

Рисунок 4.16 – Пример заполнения БД

Интеграция обучающей компоненты в игру заключается в объединении игровой компоненты и системы диалогов (окна редактирования программного кода и информационного окна). Для решения данной задачи использовался метод триггеров, который позволил реализовать гибкую систему взаимодействия между набором игровых решений и модулем адаптации.

Основная сложность реализации заключалась в учете всех вариантов выбора стратегий обучения пользователем. При реализации ad-hoc версии игры все варианты стратегий приходилось учитывать в ручную и расставлять триггеры для каждой цепочки нелинейных сюжетов.

Экранная формы игры «Камми» в режиме заполнения ответа на задание представлена на рисунке 4.17.

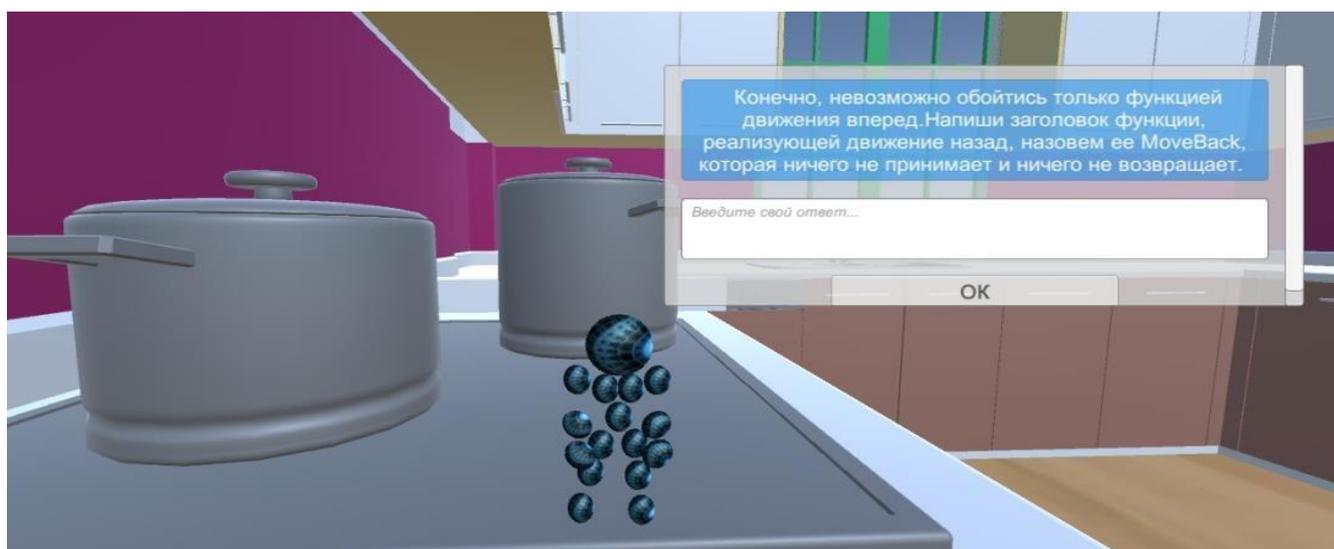


Рисунок 4.17 – Экранная форма «Обучение в форме Камми»

Экранная формы игры «Камми» в режиме игрового квеста представлена на рисунке 4.18.

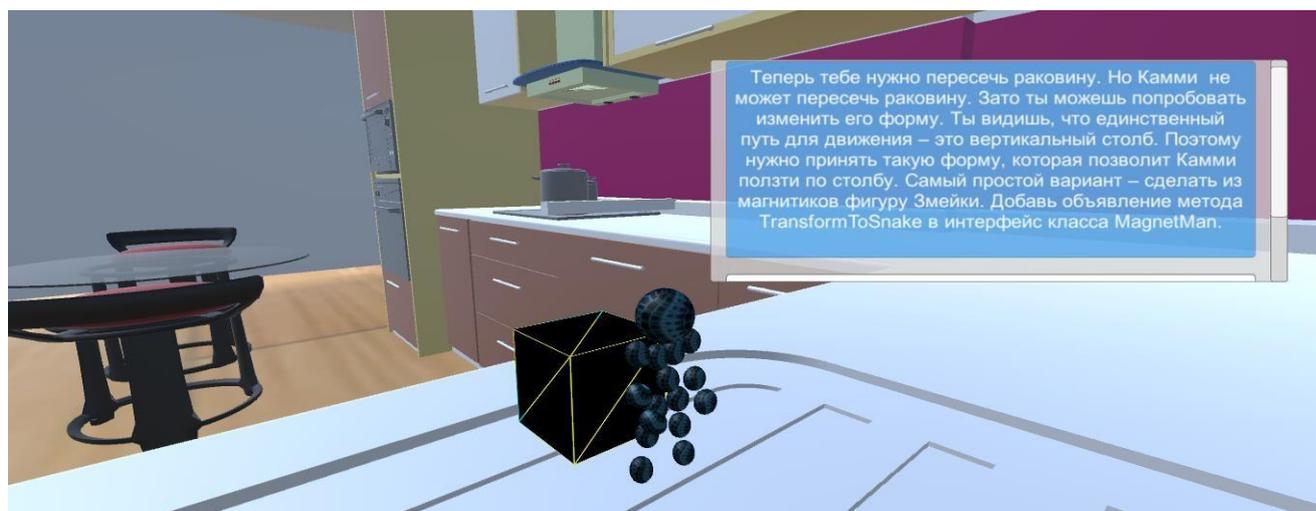


Рисунок 4.18 – Экранная форма «Квест - Черный куб»

4.3 Разработка адаптивной обучающей игры с применением метода автоматизированного проектирования

В новой версии игры «Камми» игровой мир представляется футуристической комнатой (рисунок 4.19). И все игровые текстуры представляются в новой футуристической стилистике.

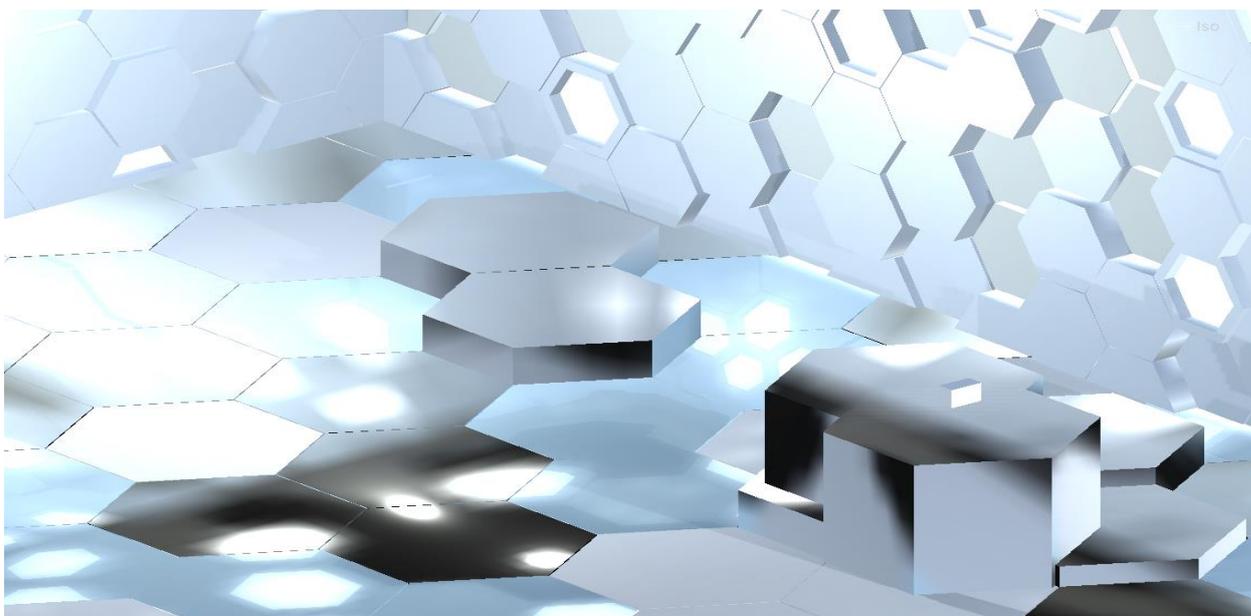


Рисунок 4.19 – Экранная форма игрового мира игра «Камту»

На рисунках 4.20 – 4.23 представлены все формы персонажа Камми в новой стилистике. Набор игровых решения используются из ad-hoc версии игры.



Рисунок 4.20 – Форма камми



Рисунок 4.21 – Форма гиря



Рисунок 4.22 – Форма змея

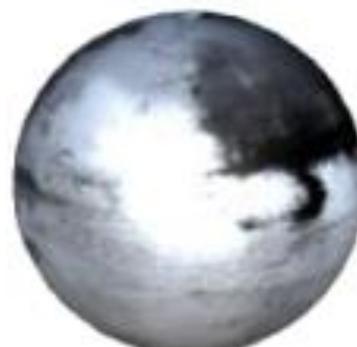


Рисунок 4.23 – Форма шар

пространства знаний в режиме выбора вариантов способа загрузки входных данных представлена на рисунке 4.25.

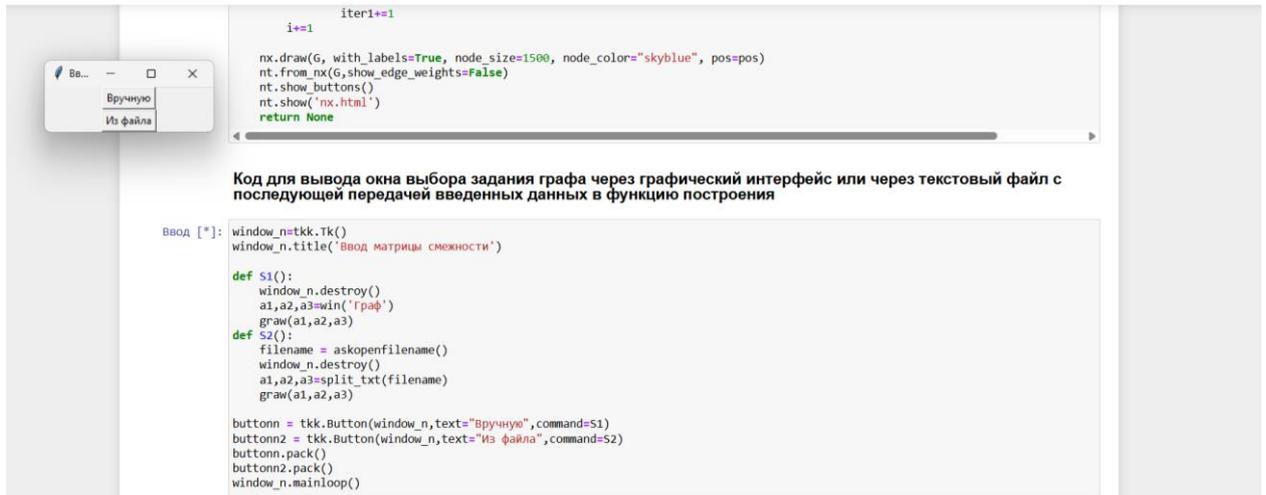


Рисунок 4.25 – Экранная форма работы инструмента для построения пространства знаний в режиме выбора способа загрузки входных данных

В результате построено пространство знаний в формате *.html* для дальнейшей работы с данным пространством. Экранная форма работы инструмента для построения пространства знаний в режиме построение пространства представлена на рисунке 4.26.

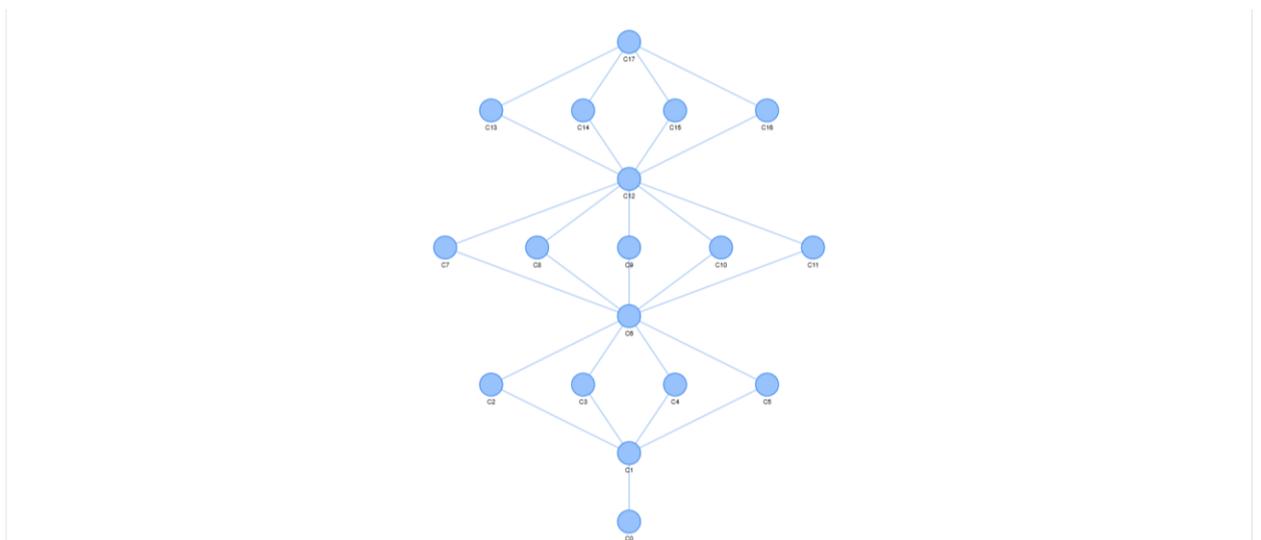


Рисунок 4.26 – Экранная форма работы инструмента для построения пространства знаний в режиме построение пространства

4.3.4 Разработка архитектуры адаптивной обучающей игры

Разработана архитектура адаптивной обучающей игры «*Kamtu*» с применением шаблона, которая учитывает архитектуру реализованного модуля адаптации (рисунок 4.27).



Рисунок 4.27 – Архитектура адаптивной обучающей игры «*Kamtu*»

4.3.5 Интеграция модуля адаптации с набором игровых решений и настройка механизмов адаптации

Произведен импорт Unity-решений путем переноса префабов в папку с проектом игры.

Весь сценарий игры представлен как совокупность элементов диалога (таблица 4.7), а элементы диалога разделены на три группы:

- текст с вариантами выбора заданий далее ТВ (вывод элемент диалога в виде текста без поля для ввода ответа на задания и кнопки для выбора задания);
- текст далее Т(вывод элемента диалога в виде текста);
- задание далее З(вывод элемент диалога в виде текста с полем для ввода ответа).

Таблица 4.7 – Фрагменты игрового сценария сопоставленные группе диалога

Группа диалога	Элемент диалога
ТВ	<p>Осмотри, сейчас ты находишься в магнитной комнате на одном из магнитных пьедесталов. В форме Камми спуститься на пол не получится, ты рассыпишься. Твоя задача перебраться на соседний пьедестал, но для этой задачи невозможно обойтись только функцией движения вперед. Тебе нужно научиться двигаться: назад, влево и вправо.</p> <p>Варианты выбора заданий:</p> <ul style="list-style-type: none"> - Научиться двигаться назад; - Научиться двигаться вперед; - Научиться двигаться вправо.
З	<p>Пробуй написать функцию движения назад MoveBack. Для движения назад нужно уменьшить координату y на 1.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит на месте; – вывод подсказки (Верный ответ y--;). <p>Верно выполнил задание: анимация шага назад.</p>
З	<p>Теперь напиши вызов функции MoveBack. Вызов функции MoveBack сопоставлен кнопке S.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит, не может двигаться по нажатию кнопки; – вывод подсказки (Верный ответ MoveBack();). <p>Выполнил верно: Камми может двигаться по нажатию кнопки.</p>
З	<p>Пробуй написать функцию движения, реализующую шаг вправо, назовем ее MoveRight. Для движения назад нужно увеличить координату x на 1.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит на месте; – вывод подсказки (Верный ответ x++;). <p>Выполнил верно: анимация шага вправо.</p>
З	<p>Напиши вызов функции. Вызов функции MoveRight сопоставлен кнопке D.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит, не может двигаться по нажатию кнопки; – вывод подсказки (Верный ответ MoveRight ();). <p>Выполнил верно: Камми может двигаться по нажатию кнопки.</p>
З	<p>Пробуй написать функцию движения, реализующую шаг влево, назовем ее MoveLeft.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит на месте; – вывод подсказки (Верный ответ x--;). <p>Выполнил верно: анимация шага влево.</p>
З	<p>Напиши вызов метода MoveLeft. Вызов метода MoveLeft сопоставлен кнопке A.</p> <p>Не верно выполнил задание:</p> <ul style="list-style-type: none"> – Камми стоит, не может двигаться по нажатию кнопки; – вывод подсказки (Верный ответ MoveLeft ();). <p>Выполнил верно: Камми может двигаться по нажатию кнопки.</p>
Т	Отлично, теперь ты можешь управлять движением Камми по всем направлениям!

В готовых решениях модуля уже присутствуют примеры заполнения для разных событий диалога (диалог без задания, диалог с заданием и окном ввода ответа, диалог с заданием и выбором ответа, игровое событие). Поэтому заполнение БД производилось для каждого из элементов диалога в соответствие с представленным шаблоном модуля адаптации, и добавлением вариантов ответов и текст подсказки.

Для текстовой информации без задания заполнен пример с атрибутом `message`. Указан атрибут `question id`, который служит идентификатором данного задания и `toQuestion` идентификатор следующего задания или диалога без задания. В атрибуте `text` заполнена информация диалога.

Для диалога с заданием заполнен пример с атрибутом `messagetwo`. Указан атрибут `question id`, который служит идентификатором данного задания и `toQuestion` идентификатор следующего задания или диалога без задания. В атрибуте `text` заполнена информация диалога. Заполнены варианты верных ответах в разметке `answer` как указано в примере, а в разметке `hint` заполнен текст подсказки, которая будет выводиться при определенном количестве неверных ответов.

Для игрового события заполнен пример с атрибутом `custom`. Указан атрибут `question id`, который служит идентификатором данного задания и `toQuestion` идентификатор следующего задания или диалога без задания. В атрибуте `gameObjectName` заполнено имя объекта в сцене к которому привязан класс `TestScene`, а в атрибуте `methodName` указано название метода.

Для удобства каждый раздел обучающего курса описан в отдельном БД файле. Пример фрагмента, заполненного БД с учетом группы диалогов игрового сценария и идентификатором пространства знаний показан на рисунке 4.28.

```

<question id="2" text=" Теперь нужно разработать алгоритм функции MoveForward. Положение любого объекта на плоскости определяется двумя
целочисленными координатами: x, y. Для того чтобы переместиться на один шаг, нужно изменить соответствующую координату на +1 или -1.
Будем считать, что по отношению к Камми ось x направлена вправо, ось y – вперед. Тогда шаг вперед эквивалентен увеличению координаты y
на единицу: " toQuestion="3">
  <answer>
    <![CDATA[ y=y+1; ]]>
  </answer>
  <answer>
    <![CDATA[ y++; ]]>
  </answer>
  <hint>y++; y=y+1; </hint>
</question>
<question id="3" text=" Для того, чтобы функция заработала, ее нужно вызвать. Вызов функции выглядит так: имя_функции (список
фактических параметров); Напиши вызов функции MoveForward. " toQuestion="4" type="messagetwo">
  <answer>
    <![CDATA[ MoveForward (); ]]>
  </answer>
  <hint>MoveForward ();</hint>
</question>
<question id="5" text="Молодец,ты научил Камми ходить вперед. Попробуй подвигаться удерживая клавишу - W " toQuestion="6"
type="message"> </question>
<question id="7" exit="true" type="custom" gameObjectName="triggerC0" className="TestScene" methodName="triggerC0"/>
<question id="8" text=" Конечно, невозможно обойтись только функцией движения вперед.Напиши заголовок функции, реализующей движение
назад, назовем ее MoveBack, которая ничего не принимает и ничего не возвращает. " toQuestion="9" type="messagetwo">
  <answer>
    <![CDATA[ 1 ]]>
  </answer>
  <answer>
    <![CDATA[ [void MoveBack(); ]]>
  </answer>
  <hint>Player player.name= '';</hint>

```

Рисунок 4.28 – Фрагмент заполненного БД игрового сценария

В модуле адаптации присутствуют несколько вариантов триггеров для активации системы диалогов, для реализации игры использованы варианты при столкновении с объектом (видимые и невидимые для пользователя). В сцену добавлены триггеры и расставлены в соответствии с пространством знаний. В соответствии с построенным пространством в сцену скопировано 17 триггеров.

Настроены свойства для каждого из триггеров (рисунок 4.29), выбран игровой персонаж, количество файлов БД и их имена, id диалога задания или игрового события в соответствии с игровым сценарием.

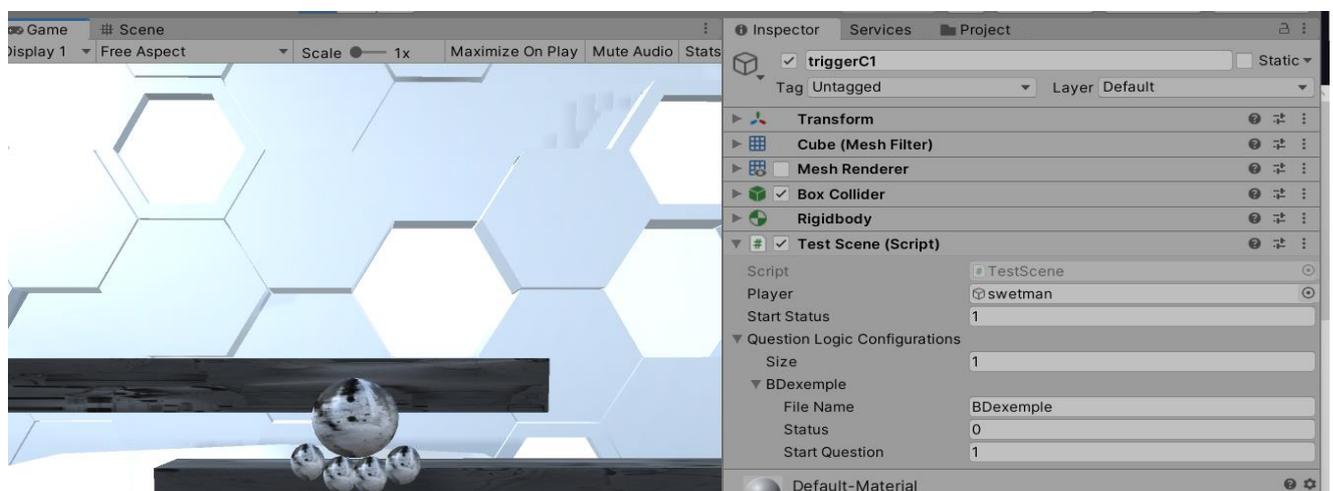


Рисунок 4.29 – Пример настройки свойств триггера

В сцену добавлена система диалоговых окон и настроены необходимые параметры. Так как в системе диалогов уже присутствовали все нужные типы окон, были заданы только необходимые параметры: количество неверных ответов для вывода подсказки, необходимые размеры и стиль, который подходит под футуристическую стилистику (рисунок 4.30).

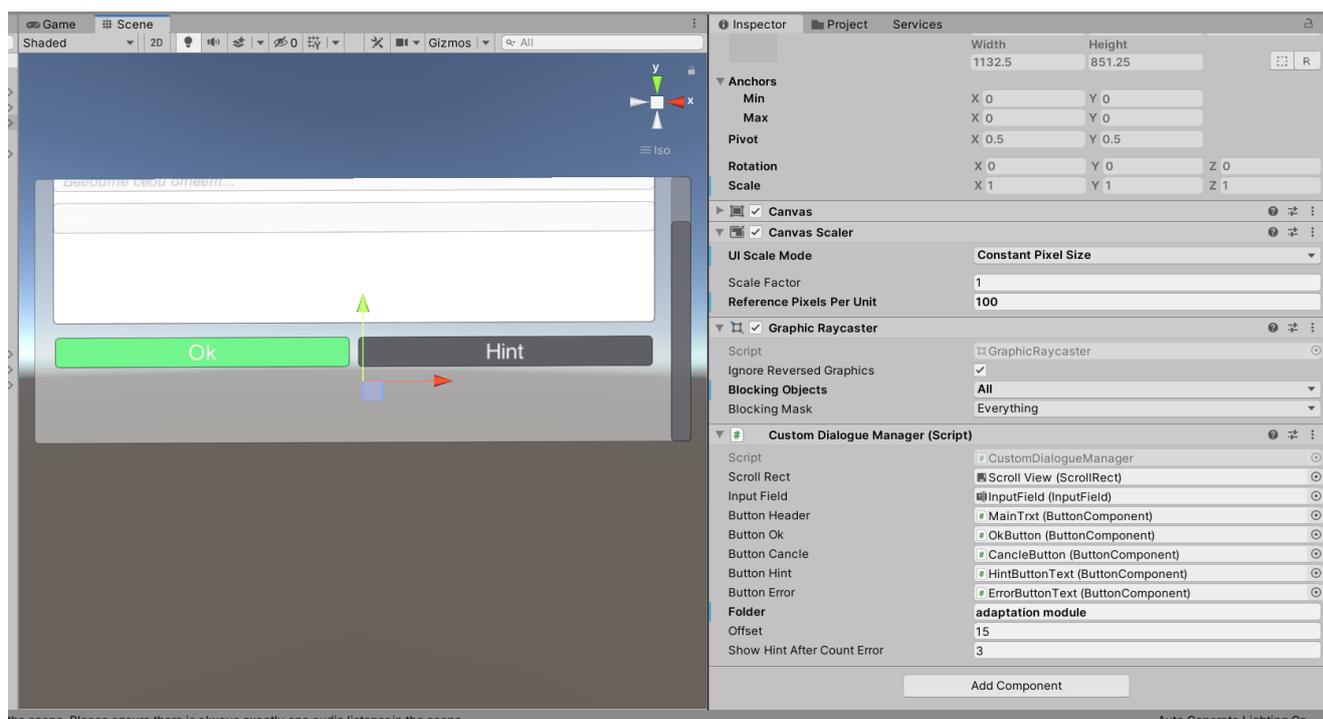


Рисунок 4.30 – Пример настройки свойств системы диалоговых окон

4.4 Апробация метода автоматизированного проектирования на других адаптивных обучающих играх

Предложенный метод автоматизированного проектирования адаптивных обучающих игр был апробирован на двух других адаптивных обучающих играх.

Адаптивная обучающая игра «Кулинар» предназначена для обучения основам кулинарии, разработана в ходе бакалаврской работы студентом ВолГТУ. Игра реализована на Unity3d и для разработки игры был использован модуль адаптации. Игра «Кулинар» в режимах обучающего задания и игрового задания представлена рисунках 4.31 и 4.32.

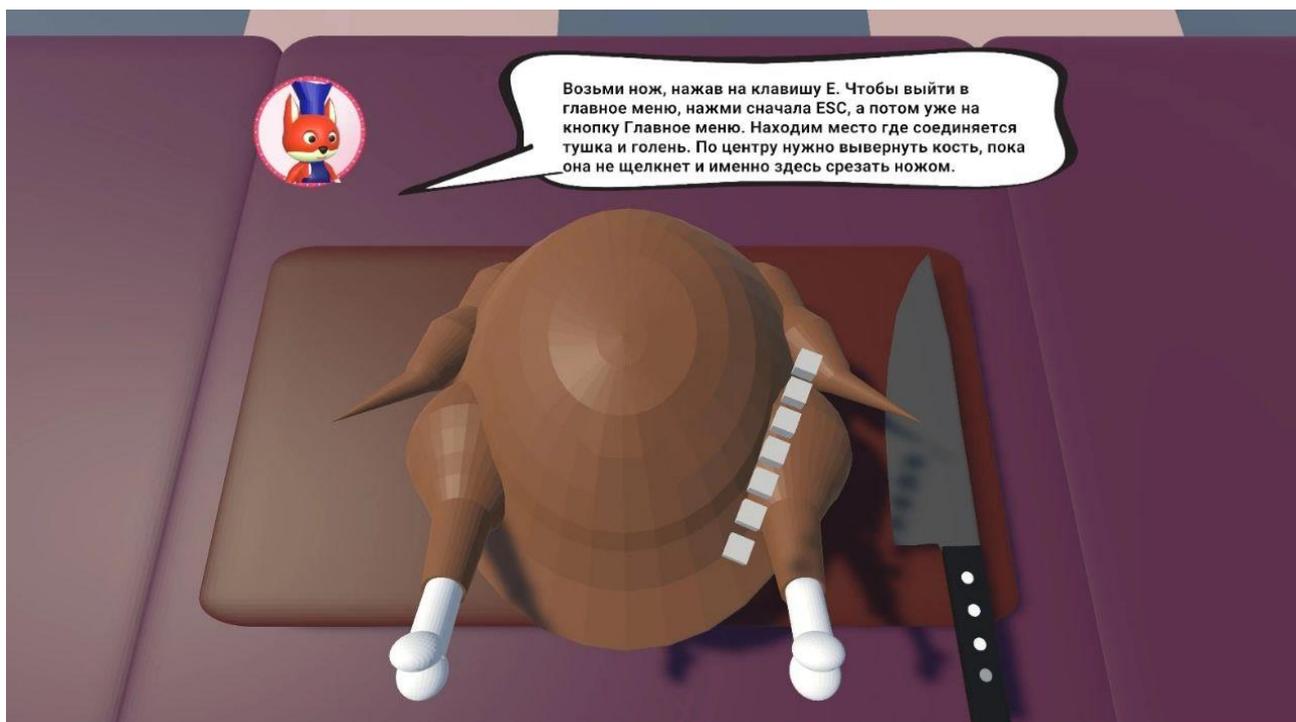


Рисунок 4.31 – Игра «Кулинар» в режиме обучающего задания



Рисунок 4.32 – Игра «Кулинар» в режиме игрового задания

Адаптивная обучающая игра «*Frog*» предназначена для обучения основам алгоритмов, разработана в рамках дисциплины «Учебная технологическая (проектно-технологическая) практика» группой студентов ВолГТУ. Игра

реализована на движке Godot с применением предложенного метода автоматизированного проектирования. Платформозависимая часть модуля реализовывалась в соответствии с логикой предложенной модели адаптации. Игра «*Frog*» в режимах обучающего задания и игрового задания представлена рисунках 4.33 и 4.34.

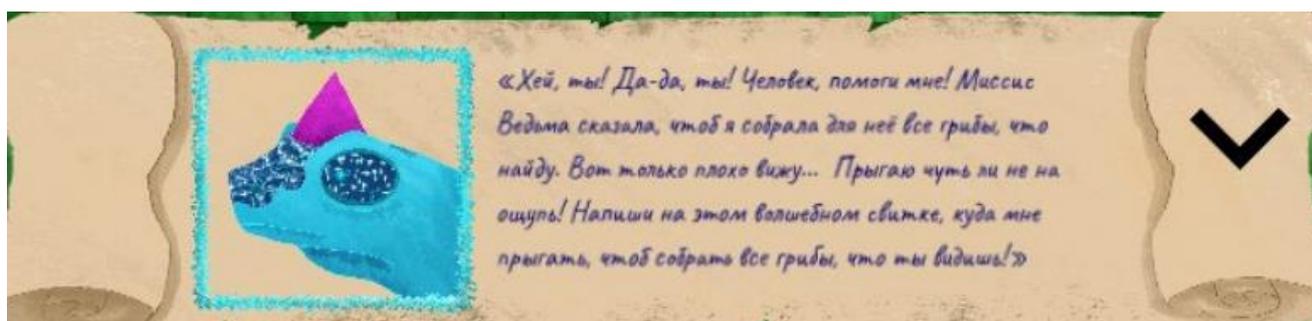


Рисунок 4.33 – Игра «*Frog*» в режиме обучающего задания



Рисунок 4.34 – Игра «*Frog*» в режиме игрового задания

4.5 Выводы по главе

В Главе 4 описана апробация предложенных моделей, методов и технологий на разработке трех адаптивных обучающих игр:

- двух игр на движке *Unity* и одной игры на движке *Godot*.
- описан неадаптивный прототип игры для обучения технологии объектно-ориентированного проектирования и языку *C#*;
- описана ad-hoc разработка адаптивной версии обучающей игры;
- описана разработка адаптивной обучающей игры с применением метода автоматизированного проектирования:
- разработана структура курса, описано построение пространства знаний и способ интерпретации обучающих действий в игровом контексте;
- разработана архитектура адаптивной обучающей игры и реализована интеграция модуля адаптации с набором игровых решений:
- для проектирования двух игр был использован шаблон архитектуры и платформонезависимый набор решений для построения пространств знаний; для реализации игр использован платформозависимый набор решений на *Unity*: системы диалоговых окон, наборы триггеров и шаблон БД;
- для третьей игры разработан платформозависимый набор решений на движке *Godot* с сохранением логики предложенной модели адаптации.

Глава 5. Оценка качества и снижение трудоемкости разработки адаптивных обучающих игр

5.1 Методы оценки качества адаптивных обучающих игр

В условиях активного развития цифровых образовательных технологий особую значимость приобретает проблема объективной оценки качества адаптивных обучающих игр. Разработка эффективных методов такой оценки представляет собой междисциплинарную задачу, требующую интеграции подходов из области педагогики, психологии, гейм-дизайна и компьютерных наук [98, 99].

В современной научной литературе выделяют три основных подхода к оценке качества АОИ: пользовательское тестирование, экспертное оценивание и автоматизированная оценка.

Пользовательское тестирование [100] представляет собой эмпирический метод оценки качества адаптивных обучающих игр, основанный на непосредственном взаимодействии представителей целевой аудитории с игровым продуктом. Данный подход позволяет получить объективные данные о эффективности учебного процесса, уровне вовлеченности пользователей, удобстве игрового интерфейса, корректности работы адаптивных механизмов.

В современной практике оценки обучающих игр применяются следующие группы методов:

1. Педагогическое тестирование [101]:
 - претесты и посттесты знаний;
 - контрольные задания;
 - анализ кривой обучения.
2. Психологическое оценивание [102]:
 - шкалы измерения мотивации (например, *intrinsic motivation inventory*);
 - методики оценки вовлеченности (*game engagement questionnaire*);
 - измерение эмоционального состояния (метод *sam*);

- юзабилити-исследования;
 - протокол «мысли вслух».
3. Eye-tracking исследования [103]:
- анализ тепловых карт;
 - игровая аналитика;
 - трекинг игровых сессий;
 - анализ поведенческих паттернов;
 - статистика использования игровых механик.

Экспертное оценивание [104, 105] представляет собой методологический подход к анализу качества адаптивных обучающих игр, осуществляемый специалистами в области педагогического дизайна, игровых технологий, пользовательского опыта и предметной области обучения, который характеризуется системностью анализа, опорой на профессиональные стандарты, использованием формализованных критериев оценки и комплексным подходом к исследованию всех компонентов игровой обучающей системы.

Экспертная оценка качества адаптивных обучающих игр осуществляется посредством комплекса взаимосвязанных методов, включающих:

1. Эвристический анализ, основанный на применении специализированных оценочных метрик для обучающих игр, или авторских методик оценки игровых компонентов с учетом критериев Nielsen-Molich [106, 107].
2. Когнитивное тестирование, направленное на выявление когнитивных нагрузок, оптимальных уровней сложности и эффективности обучающих траекторий [108].
3. Структурно-функциональный анализ, предусматривающий декомпозицию игровых механик, оценку взаимосвязей элементов системы, верификацию адаптивных алгоритмов [109].
4. Сравнительно-сопоставительный анализ, включающий оценку инновационного потенциала и анализ конкурентных преимуществ [110].

Автоматизированная оценка качества адаптивных обучающих игр реализуется через систему взаимосвязанных технологических решений, включающих [111, 112]:

- сбор и анализ игровой аналитики с использованием специализированных платформ (Unity Analytics, GameAnalytics), позволяющий отслеживать ключевые показатели эффективности обучения;
- применение алгоритмов машинного обучения для обработки больших массивов данных о поведении пользователей и автоматической корректировки учебных траекторий;
- методы stealth-аессмента, обеспечивающие непрерывную и ненавязчивую диагностику учебных достижений;
- А/В-тестирование различных вариантов игровых механик и адаптивных стратегий.

Каждый из трех основных подходов к оценке качества адаптивных обучающих игр обладает уникальными преимуществами и ограничениями.

Пользовательское тестирование обеспечивает высокую достоверность данных за счет непосредственного взаимодействия с целевой аудиторией, однако требует значительных временных и финансовых затрат, а также применимо лишь на поздних стадиях разработки.

Экспертное оценивание позволяет выявлять системные проблемы на ранних этапах проектирования, отличается гибкостью и относительно низкой стоимостью, но подвержено субъективности оценок и зависит от квалификации специалистов.

Автоматизированная оценка обеспечивает масштабируемость и объективность за счет анализа больших данных и алгоритмов машинного обучения, однако требует сложной ИТ-инфраструктуры и не всегда учитывает психолого-педагогические аспекты обучения.

5.2 Метод эвристических оценок качества для оценки качества обучающего и адаптивного обучающего ПО

Метод эвристических оценок, первоначально разработанный для оценки пользовательских интерфейсов [113], претерпел значительную эволюцию при применении к обучающему и адаптивному обучающему программному обеспечению, приобретя следующие специфические характеристики [114]:

- комплексность оценки, объединяющая педагогическую эффективность, игровую вовлеченность и техническую реализацию, что позволяет учитывать разные аспекты качества обучающих систем;
- многоуровневость анализа, включающая макроуровень (общая концепция системы), мезоуровень (игровые механики и сценарии) и микроуровень (элементы интерфейса и взаимодействия), что обеспечивает всестороннюю оценку всех компонентов системы;
- интеграцию с современными технологиями, такими как системы сбора и анализа данных, методы когнитивного тестирования и инструменты отслеживания поведения пользователей, что позволяет получать более точные и объективные показатели эффективности.

В контексте оценки качества обучающих игр важным аспектом является создание метрик, отражающих баланс между образовательной ценностью и игровой привлекательностью. В статье [115] подчеркивают, что обучающие игры должны поддерживать оптимальный уровень вызова, соответствующий навыкам игрока и выделяют метрики на основе идеи состояния потока.

Современные методы оценки адаптивных обучающих систем приобрели более комплексный характер, охватывая три ключевых направления: адаптивность на основе психологических характеристик, эмоциональных состояний и когнитивных способностей пользователей [116]. В рамках каждого направления разрабатываются метрики, позволяющие учитывать индивидуальные особенности для повышения вовлеченности и эффективности обучения. Моделирование личностных черт помогает более точно адаптировать контент, учет

эмоционального состояния способствует более длительному мотивирующему взаимодействию, а оценка когнитивных способностей позволяет динамически регулировать сложность материалов, снижая когнитивную нагрузку и повышая уровень усвоения знаний. Все эти подходы подчеркивают важность учета психологических, эмоциональных и когнитивных факторов для создания более персонализированных и эффективных образовательных систем. В рамках современных методов оценки адаптивных обучающих систем особое значение приобретает использование комплексных и мультидисциплинарных метрик, позволяющих учитывать широкий спектр факторов, влияющих на эффективность обучения и качество взаимодействия с системой [117].

В работе [118] описано как метод позволяет формализовать качественные характеристики адаптивных обучающих игр через набор метрик, объединенных в три категории, направленные на обучающую составляющую: вовлеченность пользователя, решение задач и формирование понимания. В статье [119] описан подход который предполагает, что большинство метрик для оценки качества адаптивных обучающих игр должны быть смещены сторону игровой составляющей. В данном случае набор метрик формируется из трех категорий: надежность, привлекательность и образовательная ценность. Похожий подход по отбору метрик описан в работе [120], авторы сформировали набор из 51 метрики с большим упором на игровую привлекательность.

Таким образом метод эвристических оценок качества для адаптивных обучающих игр представляет собой систематизированный подход к выявлению недостатков и преимуществ на основе экспертного анализа соответствия ключевым принципам проектирования игрового и обучающего контента.

Согласно исследованиям [121] использование комплексных метрик, охватывающих различные аспекты интерфейса и взаимодействия, позволяет выявить ключевые недостатки системы на ранних стадиях её развития. Это способствует своевременному устранению проблем, повышению качества адаптивного обучающего программного обеспечения, снижает затраты времени и ресурсов на последующие доработки.

Метод заключается в проверке удовлетворения оцениваемой адаптивной обучающей системы набору эвристически определенных характеристик, которые по мнению разработчиков обучающего ПО соответствуют требуемому уровню качества. Данный подход базируется на следующих ключевых принципах:

- эвристики формулируются как бинарные (да/нет) или шкалируемые утверждения;
- каждый критерий имеет четкое определение.

В данной работе выбор метода эвристической оценки в качестве основного инструмента оценки качества адаптивных обучающих игр обоснован совокупностью факторов, включая:

- возможность применения на ранних стадиях разработки АОИ, что соответствует принципу экономической эффективности научных исследований;
- комплексный характер оценки, позволяющий одновременно анализировать педагогические, игровые и технические аспекты адаптивных обучающих игр;
- соответствие принципам системности и объективности научного исследования, так как метод предусматривает использование стандартизированных оценочных метрик и четких критериев анализа;
- практическую значимость получаемых результатов, которые могут быть непосредственно использованы для совершенствования разрабатываемой игры.

Указанные преимущества делают эвристическую оценку оптимальным методом для решения поставленных в исследовании задач, позволяя получить достоверные данные при минимальных временных и финансовых затратах, что особенно важно в условиях ограниченных ресурсов образовательных проектов.

5.3 Адаптация метода эвристических оценок для оценки качества АОИ

Методология оценки качества адаптивных обучающих игр основана на комплексном подходе, включающем ключевые аспекты анализа, подтвержденные исследованиями. В первую очередь рассматриваются адаптивные компоненты: персонализация, корректность алгоритмов и точность диагностики знаний, обеспечивающие индивидуальный подход. Особое внимание уделяется педагогическим аспектам: соответствию стандартам, эффективности методик и объективности оценивания. Не менее важны игровые характеристики: баланс сложности, мотивационная структура и увлекательные механики, способствующие вовлеченности.

На основе анализа научных работ, посвященных оценке качества обучающего и адаптивного программного обеспечения выбраны эвристики, позволяющие оценить влияние модели адаптации на качество адаптивной обучающей игры. Всего представлено 35 эвристик (таблица 5.1), разбитых на 4 группы:

- 9 эвристик качества модели адаптации (ЭКМА);
- 10 эвристик качества структуры курса обучающих материалов (ЭКСК);
- 8 эвристик качества процесса обучения (ЭКПО);
- 8 эвристик качества механизмов оценивания и адаптации процесса обучения (ЭКМО).

Таблица 5.1 – Эвристики качества для адаптивных обучающих игр

Идентификатор	Описание
ЭКМА 1	Уровень вмешательства в процесс обучения. При явном сборе информации теряется игровая привлекательность.
ЭКМА 2	Уровень адаптации. (к знаниям или навыкам)
ЭКМА 3	Обучающий курс полностью пройден. В конце игры в независимости от стилей обучения, предпочтений, уровня знаний и навыков весь курс пройден каждым обучаемым.
ЭКМА 4	Переносимость. Модель можно реализовывать и использовать для всех жанров игр.

Продолжение таблицы 5.1

ЭКМА 5	Видимость обучения. Существует визуальная обратная связь, чтобы информировать обучаемых об их текущем прогрессе.
ЭКМА 6	Разные уровни обучаемого. Один и тот же контент представлен в различных форматах в зависимости от стиля обучения, уровня знаний и навыков.
ЭКМА 7	Помощь. Сообщение об ошибках и вывод подсказок.
ЭКМА 8	Справка. Обучаемый имеет возможность открыть справку и прочитать подробную информацию о нужном ему разделе курса.
ЭКМА 9	Метод оценивания. Обучаемый видит текущий результат успеваемости, в конце осуществляется отчет.
ЭКСК 1	Различных типы обучающей активности сбалансированы, нет перекоса в сторону изучения/тестирования или иных видов активности в процессе обучения.
ЭКСК 2	Обучающий курс воспринимается целостно.
ЭКСК 3	Игрок видит структуру курса и знает место каждого обучающего действия в процессе обучения.
ЭКСК 4	Необходимость изучения материала для достижения игровых целей осознается игроком.
ЭКСК 5	Имеется возможность выбора порядка изучения материала и выполнения заданий.
ЭКСК 6	Знания и навыки дисциплины приобретаются до того, как они понадобятся при выполнении заданий.
ЭКСК 7	Все приобретённые знания и навыки требуются в процессе игры.
ЭКСК 8	Теоретические материалы качественные, т.е. понятны, легко воспринимаются и достаточно подробно описывают суть явлений и знаний предметной области.
ЭКСК 9	Задания понятны. Игроку не требуется много времени, что бы понять что от него требуется в обучающих заданиях.
ЭКСК 10	Задания достаточно простые, что бы игрок мог их решить, но не слишком сложные, что бы игроку было интересно их решать.
ЭКМО 1	Игрок понимает, за что он получил свои оценки.
ЭКМО 2	Игрок видит текущие оценки результатов обучения.
ЭКМО 3	Игрок видит, какими знаниями и навыками он уже обладает и какие он может получить в дальнейшем.
ЭКМО 4	У обучаемого есть возможность повесить свои оценки знаний, выполнив задания повторно либо выполнив другие задания по той же теме.
ЭКМО 5	Оценки результатов обучения соответствуют текущему состоянию игрока.

Продолжение таблицы 5.1

ЭКМО 6	Уровень сложности заданий меняться в зависимости от индивидуальных результатов игрока.
ЭКМО 7	Обучающая система пытается определить когнитивный стиль обучаемого и подстраивается под него, предоставляя разное количество заданий и теоретических материалов или меняя их представление.
ЭКМО 8	Если обучаемый не может решить задачу с 1-2 попыток – система помогает ему, давая дополнительные материалы или более простую задачу или подсказку.
ЭКПО 1	В самом начале игры система даёт обучаемому понять, что для успешного существования в виртуальном мире ему придётся обучаться.
ЭКПО 2	При неправильном решении задания или выполнении неправильного обучающего действия игроку даётся пояснение его ошибок и предлагаются способы их устранения в дальнейшем.
ЭКПО 3	Система корректно определяет правильность решений игрока в большинстве случаев. Если решение выглядит странным или изощрённым – система корректно на него реагирует.
ЭКПО 4	Игроку предоставлены в самом начале обучения базовые навыки, необходимые для самостоятельного поиска информации.
ЭКПО 5	Выполнение обучающих заданий не требует от обучаемого использования внешней справочной системы или учебника.
ЭКПО 6	Обучаемый всегда имеет доступ ко всем изученным ранее материалам и выполненным заданиям.
ЭКПО 7	Доступный материал всегда предоставляется игроку в одинаковом виде.
ЭКПО 8	Материал, отмеченный как изученный и задания, являющиеся выполненными, останутся такими навсегда. Система может попросить обучаемого повторить их выполнения, явно указав, что происходит повторное изучение.

5.4 Сравнительная оценка качества адаптивной обучающей игры

Произведена сравнительная оценка качества (таблица 5.2) адаптивной обучающей игры «*Катту*» с применением разработанной модели адаптации и пяти

адаптивных обучающих игр: *Auto Thinking*, *Elektra*, *Minerva*, *Code Red*, *Slippery Rock Falls* по всем группам эвристик, выбранных для оценки качества.

Таблица 5.2 – Сравнительная оценка качества адаптивных обучающих игр

Эвристика	<i>Kammy</i>	<i>Auto Thinking</i>	<i>Elektra</i>	<i>Minerva</i>	<i>Code Red</i>	<i>Slippery Rock</i>
ЭКМА 1	1	1	1	0	1	1
ЭКМА 2	1	0	1	0	1	1
ЭКМА 3	1	1	1	1	0	0
ЭКМА 4	1	0	0	0	0	0
ЭКМА 5	1	1	1	1	0	1
ЭКМА 6	0	0	0	1	1	0
ЭКМА 7	1	1	1	0	0	0
ЭКМА 8	1	0	0	0	0	0
ЭКМА 9	1	1	1	1	1	1
ЭКСК 1	1	1	1	1	1	1
ЭКСК 2	1	1	1	1	0	0
ЭКСК 3	0	0	0	0	1	1
ЭКСК 4	1	1	1	1	1	1
ЭКСК 5	1	0	0	0	1	0
ЭКСК 6	1	0	1	0	1	1
ЭКСК 7	1	0	1	0	0	1
ЭКСК 8	1	1	1	1	1	1
ЭКСК 9	1	1	1	1	1	1
ЭКСК 10	1	1	1	1	1	1
ЭКМО 1	1	1	1	1	1	1
ЭКМО 2	1	1	1	1	1	1
ЭКМО 3	1	1	1	1	0	1
ЭКМО 4	1	1	1	1	1	1
ЭКМО 5	1	1	1	1	1	1
ЭКМО 6	0	0	0	0	1	0
ЭКМО 7	0	0	0	1	0	0
ЭКМО 8	1	1	1	0	1	0
ЭКПО 1	1	1	1	1	1	1
ЭКПО 2	1	1	1	0	0	0
ЭКПО 3	1	1	1	1	1	1
ЭКПО 4	1	0	1	1	0	1
ЭКПО 5	1	1	1	1	0	1
ЭКПО 6	1	0	0	0	0	0
ЭКПО 7	1	1	1	0	1	1
ЭКПО 8	1	1	1	1	1	1

Оценка по каждой отдельной эвристике осуществлялась в баллах на бинарной шкале [0;1] в зависимости от того, обеспечивает или нет модель соответствующее качество (рисунки 5.1 – 5.4).

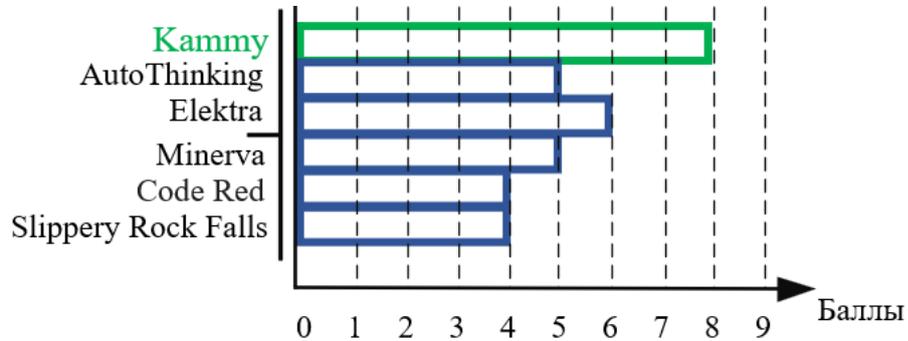


Рисунок 5.1 – Оценка качества модели адаптации (ЭКМА)

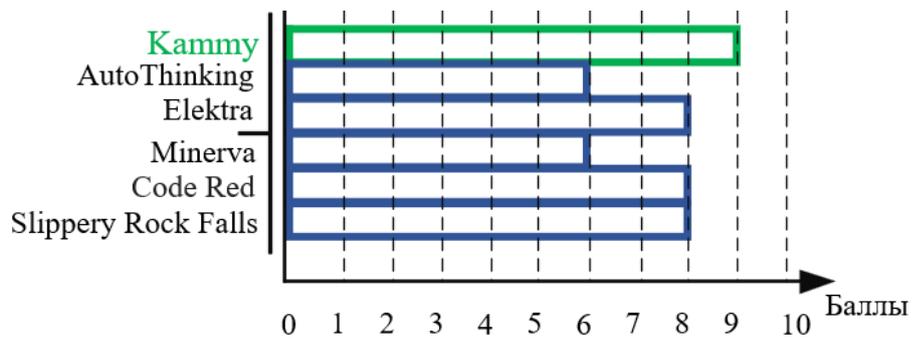


Рисунок 5.2 – Оценка структуры курса обучающих материалов

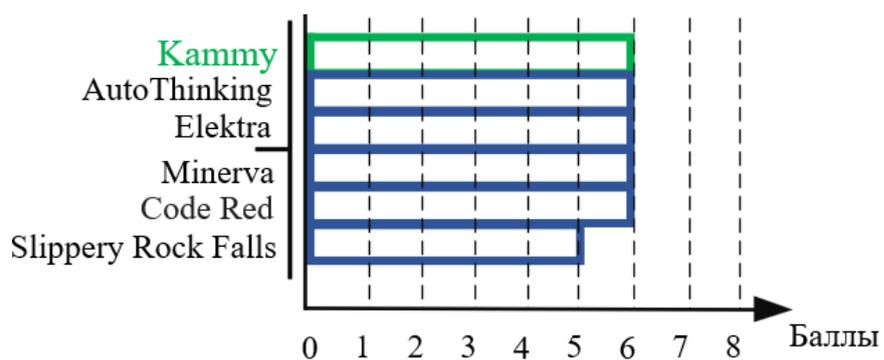


Рисунок 5.3 – Оценка качества процесса обучения

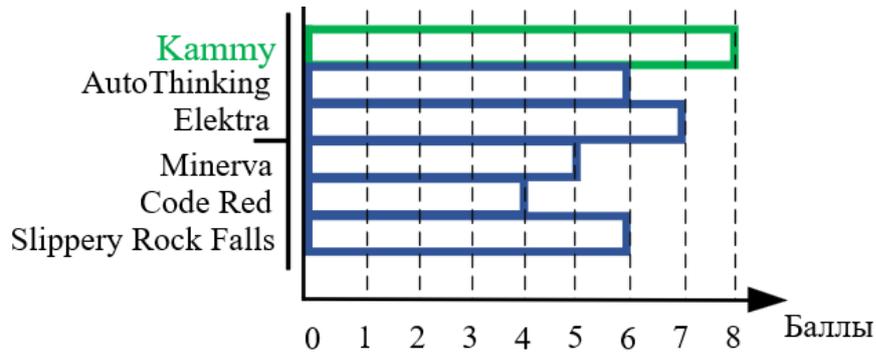


Рисунок 5.4 – Оценка качества адаптации процесса обучения

Качество моделей по каждой группе эвристик определялось как отношение набранной суммы баллов по всем эвристикам к максимально возможному значению. Результаты анализа показали, что применение предложенной модели адаптации позволило повысить качество адаптивной обучающей игры «*Kammy*» на 14.8% (31 балл из максимально возможных 35 баллов) в сравнении с игрой «*Elektra*» набравшей большее количество баллов в сравнении с другими играми (27 баллов из максимально возможных 35 баллов) (рисунок 5.5).

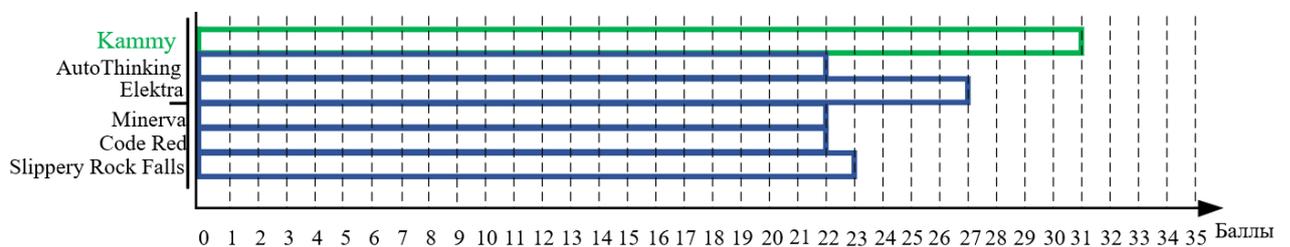


Рисунок 5.5 – Суммарная оценка качества адаптивных обучающих игр

5.5 Оценка временных затрат

Произведена оценка временных затрат на разработку адаптивных обучающих игр «*Kammy*», «*Кулинар*» и «*Frog*».

Оценка временных затрат на разработку адаптивной версии игры «*Kammy*» с применением предложенной модели адаптации в сравнении с ad-hoc разработкой прототипа игры с линейным сюжетом приведена в таблице 5.3

Таблица 5.3 – Трудоемкость разработки игры «Катту»

Этапы		<i>Ad-hoc</i> разработка	С применением метода
Этап проектирования	Разработка обучающего курса	12 ч.	12 ч.
	Разработка игрового сценария	7 ч.	7 ч.
	Разработка модели адаптации	10 ч.+	0
	Интеграция обучающего и игрового сценария с моделью адаптации	-	2 ч.
	Разработка архитектуры	2 ч.	1 ч.
Этап реализации	Реализация набора игровых решений	50 ч.	50 ч.
	Реализация модели адаптации	10 ч	0 ч.
	Импорт модуля адаптации	-	0.5 ч.
	Интеграция модели адаптации с набором игровых решений	3 ч.	2 ч.
	Настройка механизмов адаптации	2	1
Общее время разработки		96 ч.	75,5 ч.

Оценка временных затрат на разработку адаптивной версии игры «Кулинар» с применением предложенной модели адаптации в сравнении с ad-hoc разработкой прототипа игры с линейным сюжетом приведена в таблице 5.4.

Таблица 5.4 – Трудоемкость разработки игры «Кулинар»

Этапы		<i>Ad-hoc</i> разработка	С применением метода
Этап проектирования	Разработка обучающего курса	11 ч.	11 ч.
	Разработка игрового сценария	6.5 ч.	6.5 ч.
	Разработка модели адаптации	10 ч.+	0

Продолжение таблицы 5.4

Этапы		<i>Ad-hoc</i> разработка	С применением метода
Этап проектирования	Интеграция обучающего и игрового сценария с моделью адаптации	-	4 ч.
	Разработка архитектуры	2 ч.	1 ч.
Этап реализации	Реализация модели адаптации	12 ч	0 ч.
	Импорт модуля адаптации	-	2 ч.
	Интеграция модели адаптации с набором игровых решений	6 ч.	11.5 ч.
	Настройка механизмов адаптации	4	3
Общее время разработки		80 ч.	67,5 ч.

Оценка временных затрат на разработку адаптивной версии игры «*Frog*» с применением предложенной модели адаптации в сравнении с *ad-hoc* разработкой прототипа игры с линейным сюжетом приведена в таблице 5.5.

Таблица 5.5 – Трудоемкость разработки игры «*Frog*»

Этапы		<i>Ad-hoc</i> разработка	С применением метода
Этап проектирования	Разработка обучающего курса	5 ч.	5 ч.
	Разработка игрового сценария	3 ч.	3 ч.
	Разработка модели адаптации	10 ч.+	0
	Интеграция обучающего и игрового сценария с моделью адаптации	-	2,5 ч.
	Разработка архитектуры	2 ч.	1 ч.
Этап реализации	Реализация набора игровых решений	14 ч.	14 ч.
	Реализация модели адаптации	13 ч.	12 ч.
	Импорт модуля адаптации	-	-
	Интеграция модели адаптации	2 ч.	2 ч.
	Настройка механизмов адаптации	1	1
Общее время разработки		50 ч.	41,5 ч.

Трудоемкость разработки трех адаптивных обучающих игр с применением предложенной технологии проектирования в сравнении с *ad-hoc* разработкой снизилась: для игры «*Kammy*» – на 20,5 часов (21%), для игры «*Кулинар*» – на 13,5 часов (17,5%), для игры «*Frog*» на 8,5 часов (16 %) (рисунок 5.6). В целом, трудоемкость проектирования адаптивных обучающих игр с применением предложенной модели адаптации может сильно различаться в зависимости от сложности механики игры, количества игровых событий, специфики и жанра игры. Также на время реализации игры влияет выбор движка, на котором она будет реализована.

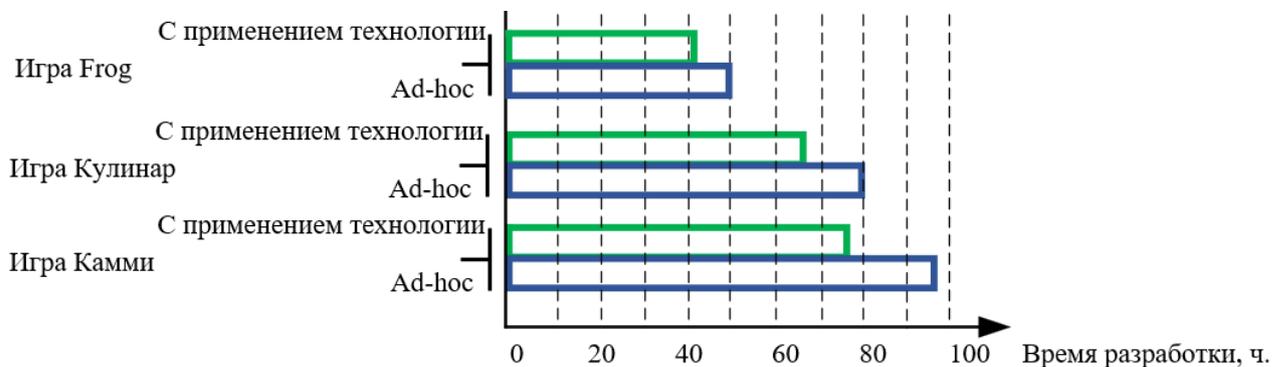


Рисунок 5.6 – Трудоемкость проектирования адаптивных обучающих игр

5.6 Выводы по главе

В Главе 5 произведена оценка качества разработанных адаптивных обучающих игр и временных затрат на их разработку:

- для оценки качества разработанных адаптивных обучающих игр применен метод эвристических оценок, разработанный для оценки качества адаптивных обучающих систем, основанный на проверке удовлетворения оцениваемой адаптивной обучаемой системы набору эвристически определенных характеристик, которые по мнению разработчиков обучающего ПО соответствуют требуемому уровню качества;

- выбраны эвристики, позволяющие оценить влияние модели адаптации на качество адаптивной обучающей игры; всего представлено 35 эвристик, разбитых на 4 группы: эвристики качества модели адаптации, эвристик качества структуры курса обучающих материалов; эвристик качества процесса обучения и эвристика качества механизмов оценивания и адаптации процесса обучения;
- произведена сравнительная оценка качества адаптивной обучающей игры «*Kamtu*» с применением разработанной модели адаптации и пяти адаптивных обучающих игр по всем группам эвристик, выбранных для оценки качества;
- показано, что применение предложенной модели адаптации позволило повысить качество адаптивной обучающей игры «*Kamtu*» на 14.8% в сравнении с игрой, набравшей большее количество баллов в сравнении с другими играми;
- произведена оценка временных затрат на разработку адаптивных обучающих игр и показано, что при разработке адаптивной обучающей игры «*Камми*» на этапе проектирования время снизилось 29% за счет использования готовой модели адаптации, а этап реализации сократился 18% за счет использования готового набора решений;
- трудоемкость разработки трех адаптивных обучающих игр с применением предложенной технологии проектирования в сравнении с *ad-hoc* разработкой снизилась: для игры «*Kamtu*» – на 21%, для игры «*Кулинар*» – на 17,5%, для игры «*Frog*» на 16 %.

Заключение

Главным результатом диссертационной работы является повышение качества адаптивных обучающих игр и снижение трудоемкости их разработки за счет нового метода автоматизированного проектирования адаптивных обучающих игр.

В ходе исследования решены следующие основные задачи:

- проведен анализ существующих моделей, методов и технологий, применяемых в сфере разработки адаптивного обучающего ПО, и возможности их применения к разработке адаптивных обучающих игр;
- разработана переносимая модель адаптации, применимая для проектирования адаптивных обучающих игр различных жанров;
- разработана технология автоматизированного проектирования адаптивных обучающих игр на основе приложенной модели адаптации;
- разработан метод, который обеспечивает возможность взаимодействия разработчиков обучающего курса и адаптивной обучающей игры для разработки модели предметной области в игровом контексте;
- формализованы проектные и технологические процедуры, которые автоматизируют процесс проектирования адаптивных обучающих игр;
- методология оценки качества адаптивных обучающих игр основана на комплексном подходе, включающем ключевые аспекты анализа, подтвержденные исследованиями. В первую очередь рассматриваются адаптивные компоненты: персонализация, корректность алгоритмов и точность диагностики знаний, обеспечивающие индивидуальный подход. Особое внимание уделяется педагогическим аспектам: соответствию стандартам, эффективности методик и объективности оценивания;
- на основе анализа научных работ, посвященных оценке качества обучающего и адаптивного программного обеспечения выбраны

эвристики, позволяющие оценить влияние модели адаптации на качество адаптивной обучающей игры;

- произведена сравнительная оценка качества адаптивной обучающей игры «*Katmu*» с применением разработанной модели адаптации и пяти адаптивных обучающих игр по всем группам эвристик, выбранных для оценки качества;
- трудоемкость разработки трех адаптивных обучающих игр с применением предложенной технологии проектирования в сравнении с *ad-hoc* разработкой снизилась: для игры «*Katmu*» – на 20,5 часов (21%), для игры «*Кулинар*» – на 13,5 часов (17,5%), для игры «*Frog*» на 8,5 часов (16 %) (рисунок 5.6). В целом, трудоемкость проектирования адаптивных обучающих игр с применением предложенной модели адаптации может сильно различаться в зависимости от сложности механики игры, количества игровых событий, специфики и жанра игры. Также на время реализации игры влияет выбор движка, на котором она будет реализована.

Таким образом, в ходе выполнения диссертационной работы были решены все поставленные задачи, цель достигнута.

В качестве перспектив дальнейшей работы можно выделить модификацию модели адаптации к стилю и предпочтениям обучаемого; расширение возможностей модуля адаптации, в том числе реализацию платформозависимых решений для других популярных движков разработки игр, а также интеграцию в модуль адаптации методов искусственного интеллекта для генерации диалогов на основе построенного курса.

Список литературы

1. Wegerif R. The role of educational software as a support for teaching and learning conversations // Computers & Education. 2004. Т. 43. С. 179–191. DOI: 10.1016/j.compedu.2003.12.012.
2. Алисултанова Э.Д., Ларснукеева М.А. Современное программное обеспечение образовательной сферы // Аллея науки. 2017. Т. 5, № 16. С. 972–976. EDN: YOOCBW.
3. Tahir M., Siddique Z., Awan S. The effectiveness of educational software and applications for primary students' learning outcomes: A systematic review // Journal of Excellence in Social Sciences. 2024. Т. 3. DOI: 10.69565/jess.v3i4.347.
4. Farooqi M. T. K., Siddique Z., Awan S. M. The effectiveness of educational software and applications for primary students' learning outcomes: A systematic review // Journal of Excellence in Social Sciences. 2024. Т. 3, № 4. С. 26–42. DOI: 10.69565/jess.v3i4.347.
5. Наземнов Д.А., Наземнова И.О., Витушкин Д.О. Изучение информационных технологий и использование современного компьютерного программного обеспечения // Актуальные исследования. 2023. № 28 (158). С. 34–37. URL: <https://apni.ru/article/6717-izuchenie-informatsionnykh-tekhnologij-i-ispolzovanie-sovremennogo-kompyuternogo-programmnogo-obespecheniya> (дата обращения: 20.04.2025).
6. Johnson L., Schleyer T. Developing High-Quality Educational Software // Journal of Dental Education. 2003. Т. 67. С. 1209–1220. DOI: 10.1002/j.0022-0337.2003.67.11.tb03712.x.
7. Liu M., Yu D. Towards intelligent E-learning systems // Educ Inf Technol. 2023. Т. 28. С. 7845–7876. DOI: 10.1007/s10639-022-11479-6.
8. Hinostroza J., Rehbein L., Mellar H., Preston C. Developing Educational Software: a professional tool perspective // Education and Information Technologies. 2000. Т. 5. С. 103–117. DOI: 10.1023/A:1009699417462.
9. Косаев К.Е. Интеллектуальные обучающие системы // Теория и практика современной науки. 2016. № 2 (8).

- URL: <https://cyberleninka.ru/article/n/intellektualnye-obuchayuschie-sistemy-2> (дата обращения: 20.04.2025).
10. Хачатурова С.С. Обучающие системы в образовании // Международный журнал экспериментального образования. 2017. № 3-2. С. 188–189.
11. URL: <https://expeducation.ru/ru/article/view?id=11285> (дата обращения: 20.04.2025).
12. Shvets O., Murtazin K., Meeter M., Piho G. Towards a Domain Model for Learning and Teaching // 2024. С. 288–296. DOI: 10.5220/0012471400003645.
13. Weber G. Adaptive Learning Systems // In: Seel N.M. (ed.) Encyclopedia of the Sciences of Learning. Boston: Springer, 2012. DOI: 10.1007/978-1-4419-1428-6534.
13. Wilson C., Scott B. Adaptive systems in education: a review and conceptual unification // Int J Inf Learn Technol. 2017. Т. 34. С. 2–19. DOI: 10.1108/IJILT-09-2016-0040.
14. Добрица В.П., Горюшкин Е.И. Применение интеллектуальной адаптивной платформы в образовании // Auditorium. 2019. № 1 (21). URL: <https://cyberleninka.ru/article/n/primenenie-intellektualnoy-adaptivnoy-platformy-v-obrazovanii> (дата обращения: 20.04.2025).
15. Contrino M.F., Reyes-Millán M., Vázquez-Villegas P. et al. Using an adaptive learning tool to improve student performance and satisfaction in online and face-to-face education for a more personalized approach // Smart Learn Environ. 2024. Т. 11, № 6. DOI: 10.1186/s40561-024-00292-y.
16. Ерогова И.С. Компьютерные игры в обучении // Вестник науки и творчества. 2016. № 9 (9). URL: <https://cyberleninka.ru/article/n/kompyuternye-igry-v-obuchenii> (дата обращения: 21.04.2025).
17. Zokirov S., Rakhmatova G. Использование компьютерных игр в современном образовании: проблемы и решения // 2023. № 1. С. 27–34. DOI: 10.5281/zenodo.7596023.
18. Jong M.S.Y., Lee J.H.M., Shang J. Educational Use of Computer Games: Where We Are, and What's Next // In: Huang R., Kinshuk S., Spector J.M. (eds) Reshaping Learning. New Frontiers of Educational Research. Berlin: Springer, 2013. DOI: 10.1007/978-3-642-32301-013.

19. Lopes R., Bidarra R. Adaptivity Challenges in Games and Simulations: A Survey // IEEE Trans Comput Intell AI Games. 2011. Т. 3. С. 85–99. DOI: 10.1109/TCIAIG.2011.2152841.
20. Chiotaki D., Pouloupoulos V., Karpouzis K. Adaptive game-based learning in education: a systematic review // Front Comput Sci. 2023. Т. 5. DOI: 10.3389/fcomp.2023.1062350.
21. Adcock A., Van Eck R. Adaptive Game-Based Learning // In: Seel N.M. (ed.) Encyclopedia of the Sciences of Learning. Boston: Springer, 2012. DOI: 10.1007/978-1-4419-1428-64.
22. Шабалина О.А., Давтян А.Г., Катаев А.В., Алимов А.А. Адаптивные обучающие игры как тренд развития обучающего ПО // ИТНОУ: Информационные технологии в науке, образовании и управлении. 2018. № 4 (8). С. 11–16.
23. Fiqri M., Nurjanah D. Graph-based domain model for adaptive learning path recommendation // IEEE Global Eng. Educ. Conf. (EDUCON). Athens, Greece, 2017. С. 375–380. DOI: 10.1109/EDUCON.2017.7942875.
24. Alevan V., Rowe J., Huang Y., Mitrovic A. Chapter 7: Domain modeling for AIED systems with connections to modeling student knowledge // In: Электронный ресурс. URL: <https://www.elgaronline.com/edcollchap/book/9781800375413/book-part-9781800375413-17.xml>.
26. Bellarhmouch Y., Jeghal A., Tairi H., Benjelloun N. A proposed architectural learner model for a personalized learning environment // Education and Information Technologies. 2022. Т. 28. С. 4243–4263. DOI: 10.1007/s10639-022-11392-y.
27. Кизелев П.А. Модели обучаемого // Эпоха науки. 2016. № 8. URL: <https://cyberleninka.ru/article/n/modeli-obuchaemogo> (дата обращения: 20.04.2025).
28. Nguyen L., Do P., Fröschl C. Learner Model in Adaptive Learning // Электронный ресурс. 2008.
- Abir A., Idrissi M., Bennani S. (2019). Learner modelling: systematic review of the literature from the last 5 years // Educational Technology Research and Development. 67. DOI: 10.1007/s11423-018-09644-1.

29. Попова, Ю. Б. Математические модели пользователей в адаптивных обучающих системах / Ю. Б. Попова, А. И. Бураковский // Информационные технологии в образовании, науке и производстве: II Международная научно-техническая интернет-конференция, 4 декабря 2014 г. Секция: Современные информационные технологии в преподавании технических и гуманитарных дисциплин [Электронный ресурс]. 2014.
30. Hu Q., Huang Y., Li Y. A “Content-Behavior” Learner Model for Adaptive Learning System // In: Huang D.S., Jo K.H., Wang L. (eds) Intelligent Computing Methodologies. ICIC 2014. Lecture Notes in Computer Science, vol 8589. Springer, Cham, 2014. DOI: 10.1007/978-3-319-09339-047.
31. Власенко А.А. Модель обучаемого в адаптивной системе обучения // Международный научно-исследовательский журнал. 2012. № 5 (5). URL: объект отсутствует (дата обращения: 20.04.2025).
32. Nguyen L., Do P., Fröschl C. Learner Model in Adaptive Learning // Электронный ресурс. 2008.
33. Lindberg R., Laine T. Detecting Play and Learning Styles for Adaptive Educational Games // In: Proceedings of the 2016 International Conference on Advanced Learning Technologies. DOI: 10.5220/0005858501810189.
34. Lindberg R., Laine T. Approaches to Detecting and Utilizing Play and Learning Styles in Adaptive Educational Games // In: Advances in Intelligent Systems and Computing, vol 978-3-319-63184-4, 2017. DOI: 10.1007/978-3-319-63184-418.
35. Seghroucheni Y., Chekour M. An Adaptive Mobile System Based on the Felder-Silverman Learning Styles Model // International Journal of Interactive Mobile Technologies (iJIM). 2022. № 16. С. 174–183. DOI: 10.3991/ijim.v16i18.34127.
36. Bontchev B., Vassileva D. Courseware Adaptation to Learning Styles and Knowledge Level // DOI: 10.5772/29340.
37. Зайцева Л.В. Методы и модели адаптации к учащимся в системах компьютерного обучения // ОТО. 2003. № 4. URL: <https://cyberleninka.ru/article/n/metody-i-modeli-adaptatsii-k-uchaschimsya-v-sistemah-kompyuternogo-obucheniya> (дата обращения: 20.04.2025).

38. Tumaini Kabudi I., Pappas I., Olsen D. H. AI-enabled adaptive learning systems: A systematic mapping of the literature // *Computers and Education: Artificial Intelligence*. 2021. T. 2. DOI: 10.1016/j.caeai.2021.100017.
39. Taylor D., Yeung M., Bashet A.Z. Personalized and Adaptive Learning // DOI: 10.1007/978-3-030-58948-62.
40. Pavlov R., Paneva-Marinova D. Personalized and Adaptive Learning – Approaches and Solutions.
41. Hssina B., Erritali M. A Personalized Pedagogical Objectives Based on a Genetic Algorithm in an Adaptive Learning System // *Procedia Computer Science*. 2019. № 151. C. 1152–1157. DOI: 10.1016/j.procs.2019.04.164.
42. Velázquez A., Assar S., Defude B., Moreno J. Student Learning Styles Adaptation Method Based on Teaching Strategies and Electronic Media // 2008. DOI: 10.1109/ICALT.2008.149.
43. Zarraonandia T., Díaz P., Aedo I. Modeling Games for Adaptive and Personalized Learning // In: *Advances in Intelligent Systems and Computing*, vol 978-3-662-47724-3, 2016. DOI: 10.1007/978-3-662-47724-312.
44. Plamondon K.E., Donovan M.A., Pulakos E.D., Arad S. Adaptability in the Workplace: Development of a Taxonomy of Adaptive Performance // *Journal of Applied Psychology*. 2000. T. 85. C. 612–624.
45. Barnes T. The Q-matrix method: Mining student response data for knowledge // *AAAI Workshop - Technical Report*. 2005. pp. 39–46.
46. Chen N., Kinshuk W., Chen H. Mining e-Learning domain concept map from academic articles // *Computers & Education*. 2008. T. 50. № 3. C. 1009–1021.
47. Martin B., Mitrovic A., Koedinger K.R., Mathan S. Evaluating and improving adaptive educational systems with learning curves // *User Modeling and User-Adapted Interaction*. 2011. T. 21. № 3. C. 249–283. DOI: 10.1007/s11257-010-9084-2.
48. Cen H., Koedinger K., Junker B. Learning Factors Analysis – A General Method for Cognitive Model Evaluation and Improvement // *Intelligent Tutoring Systems. ITS 2006. Lecture Notes in Computer Science*, vol 4053. Springer, Berlin, Heidelberg, 2006.

49. Chen S., Sue P.-J. Constructing concept maps for adaptive learning systems based on data mining techniques // *Expert Systems with Applications*. 2013. T. 40. № 7. С. 2746–2755.
50. Lopes R., Bidarra R. Adaptivity challenges in games and simulations: A survey // *IEEE Transactions on Computational Intelligence and AI in Games*. 2011. T. 3. № 2. С. 85–99.
51. Houlette R. Player modeling for adaptive games // *AI game programming wisdom 2*. Ham: Charles River Media, 2003.
52. Hussaan A.M., Sehaba K., Mille A. Tailoring serious games with adaptive pedagogical scenarios: A serious game for persons with cognitive disabilities // *Proceedings of the 2011 11th IEEE-11 International Conference on Advanced Learning Technologies, ICALT 2011*. 2011. С. 486–490. DOI: 10.1109/ICALT.2011.152.
53. Tumenayu O.O., Shabalina O., Kamaev V., Davtyan A. Using agent-based technologies to enhance learning in educational games // *Proceedings of the International Conference e-Learning 2014 - Part of the Multi Conference on Computer Science and Information Systems, MCCSIS 2014*. 2014. С. 149–155.
54. Tumenayu O.O., Shabalina O. Digital educational games: adopting pedagogical agent to infer learner's motivation and emotional // название статьи на русском, журнал/конференция, год.
55. Reuter C., Mehm F., Göbel S., Steinmetz R. Evaluation of adaptive serious games using playtraces and aggregated play data // *7th European Conference on Games Based Learning, ECGBL 2013*. 2013. С. 504–511.
56. Göbel S., Mehm F., Radke S., Steinmetz R. 80 Days: adaptive digital storytelling for digital educational games // *CEUR Workshop Proceedings*. 2009. № 498. С. 1–8.
57. Morizane M., Nakano Y., Shimohara K., Tanev I. Personalized adaptive strategies in human-PC learning game // *Proceedings of the ICCAS-SICE 2009 - ICROS- SICE International Joint Conference*. 2009. С. 2728–2731.
58. Hooshyar D., Lim H., Pedaste M., Yang K., Fathi M. AutoThinking: an adaptive computational thinking game // DOI: 10.1007/978-3-030-35343-841.
59. Darwesh A. Serious games in adaptive learning // *Journal of University of Human Development*. 2016. № 2. С. 418–423. DOI: 10.21928/juhd.20161225.18.

60. Peirce N., Conlan O., Wade V. Adaptive educational games: providing non-invasive personalised learning experiences // Proceedings of the 2nd IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning, DIGITEL 2008. 2008. С. 28–35. DOI: 10.1109/DIGITEL.2008.30.
61. Peirce N., Wade V. Personalised learning for casual games: the 'Language Trap' online language learning game // 4th European Conference on Games Based Learning, ECGBL 2010.
62. Oostendorp H., Spek E., Linssen J. Adapting the complexity level of a serious game to the proficiency of players // EAI Endorsed Transactions on Serious Games. 2014. № 14.
63. Lindberg R., Laine T. Formative evaluation of an adaptive game for engaging learners of programming concepts in K-12 // International Journal of Serious Games. 2018. № 5. С. 3–24. DOI: 10.17083/ijsg.v5i2.220.
64. Lindberg R., Hasanov A., Laine T. Improving play and learning style adaptation in a programming education game // Proceedings of the 2017 Conference on Games and Learning Alliance. С. 450–457.
65. Calongne C., Stricker A., Truman B., Murray J., Lavieri E. Slippery Rocks and ALGAE: a multiplayer educational roleplaying game and a model for adaptive learning game design // Proceedings of TCC 2014. 2014. С. 13–23. URL: <https://www.learntechlib.org/p/149824/>.
66. Zulpukarova D., Smanova N., Kultaeva D. Electronic textbooks as a means of modern education // Bulletin of Science and Practice. 2022. DOI: 10.33619/2414-2948/82/62.
67. Adaptive learning platforms // Электронный ресурс. URL: <https://www.edapp.com/blog/adaptive-learning-platforms/> (дата обращения: 11.02.2025).
68. Mezin H., Ayoub L. Adaptive learning algorithms and platforms: a concise overview // DOI: 10.1007/978-3-030-91738-81.
69. Kumah M. Analyzing the VARK model of pre-service teachers' PCK of learning // Journal of Education. 2022. № 45. С. 58–70.
70. Alosi Labs // Электронный ресурс. URL: <http://www.alosilabs.org> (дата обращения: 11.02.2023).

71. Rosen Y., Rushkin I., Rubin R., Munson L., Ang A., Weber G., Lopez G., Tingley D. Adaptive learning open source initiative for MOOC experimentation // DOI: 10.1007/978-3-319-93846-257.
72. Ali A.-A. Hybrid learning using Canvas LMS // *European Journal of Education and Pedagogy*. 2021. № 2. C. 27–33. DOI: 10.24018/ejedu.2021.2.6.180.
73. Vesin B., Milicevic A., Ivanovic M., Budimac Z. Applying recommender systems and adaptive hypermedia for e-learning personalization // *Computing and Informatics*. 2013. № 32. C. 629–659.
74. Walter F.-P., Haydee F.-P., Chiri P., Cruz K. Moodle in distance education // *Puriq*. 2022. № 4. DOI: 10.37073/puriq.4.417.
75. Harrison C., Loe B., Lis P., S.-G. Chris. Delivering the full potential of patient-reported assessments: computerized adaptive testing and machine learning using the open source Concerto platform // DOI: 10.2196/preprints.20950.
76. Sun X., Zhao X., Ma Y., Yuan X., He F., Feng J. Multi-behavior features based knowledge tracking using decision tree improved DKVMN // *ACM TURC*. 2019. C. 1–6. DOI: 10.1145/3321408.3322847.
77. Shi Y., Chi M., Barnes T., Price T. Code-DKT: a code-based knowledge tracing model for programming tasks // *arXiv preprint arXiv:2206.03545*. 2022. DOI: 10.48550/arXiv.2206.03545.
78. Cosyn E., Uzun H., Doble C., Matayoshi J. A practical perspective on knowledge space theory: ALEKS and its data // *Journal of Mathematical Psychology*. 2022. № 101. C. 102512. DOI: 10.1016/j.jmp.2021.102512.
79. Shum L.C., Rosunally Y., Scarle S. et al. Personalised learning through context-based adaptation in the serious games with gating mechanism // *Educ Inf Technol*. 2023. DOI: 10.1007/s10639-023-11695-8.
80. Lavieri Jr. E.D. A study of adaptive learning for educational game design // *EAI Endorsed Transactions on Serious Games*. 2014; 14(2): 273–282.
81. Shabalina O., Vorobkalov P. Development of educational computer games: learning process model and how it is integrated into the game context // *World Applied Sciences Journal (WASJ)*. 2013; 24(24): 256–267.

- 82.Шабалина О.А. Моделирование пространства знаний на основе математической решетки // Современные проблемы и пути их решения в науке, транспорте, производстве и образовании. 2012; 4(11): 87–89.
- 83.Shabalina O., Mozelius P., Malliarakis C., Tomos F., Balan C., Blackey H., Gerkushenko G. Combining game-flow and learning objectives in educational games // Proceedings of the 8th European Conference on Games Based Learning. Berlin: University of Applied Sciences HTW, 2014. С. 529–537.
- 84.Shabalina O., Vorobkalov P., Kataev A., Kravets A. Educational computer games development: methodology, techniques, implementation // Proceedings of the 2013 International Conference on Advanced ICT. 2013. С. 419–423.
- 85.Хайров А.В., Шабалина О.А., Катаев А.В. Метод динамического контентного согласования, обучающего и игрового сценариев в адаптивных обучающих играх // Моделирование, оптимизация и информационные технологии. 2020; 8(1).
- 86.Шабалина О.А. Разработка обучающих компьютерных игр: как сохранить баланс между обучающей и игровой компонентой // ОТО. 2013. № 3. URL: <https://cyberleninka.ru/article/n/razrabotka-obuchayuschih-kompyuternyh-igr-kak-sohranit-balans-mezhdu-obuchayuschey-i-igrovoy-komponentoy> (дата обращения: 12.05.2025).
- 87.Никитин П.В., Горохова Р.И., Зайков А.С. Применение компьютерных игр как фактор повышения качества обучения информатике // ОТО. 2015. № 3. URL: <https://cyberleninka.ru/article/n/primeneniye-kompyuternyh-igr-kak-faktor-povysheniya-kachestva-obucheniya-informatike> (дата обращения: 12.05.2025).
- 88.Malliarakis C., Tomos F., Shabalina O., Mozelius P., Balan C. How to build an ineffective serious game: worst practices in serious game design // Proceedings of the 9th European Conference on Games Based Learning. Norway: Nord Trøndelag University College Steinkjer, 2015. С. 338–345.
- 89.Shabalina O., Mozelius P., Malliarakis C., Tomos F., Balan C., Blackey H., Gerkushenko G. Combining game-flow and learning objectives in educational games // Proceedings of the 8th European Conference on Games Based Learning. Berlin: University of Applied Sciences HTW, 2014. С. 529–537.

90. Shabalina O., Vorobkalov P., Kataev A., Tarasenko A. 3I-approach for IT educational games development // Proceedings of the 3rd European Conference on Games-Based Learning. Graz: FH JOANNEUM University of Applied Sciences, 2009. С. 339–344.
91. Шабалина О.А., Воробкалов П.Н., Катаев А.В. 3I-подхода для разработки обучающих игр по объектно-ориентированному программированию // Вестник компьютерных и информационных технологий. 2011; 6: 46–52.
92. Шабалина О.А., Воробкалов П.Н., Катаев А.В. 3I-подход к разработке компьютерных игр для обучения техническим дисциплинам // Вестник компьютерных и информационных технологий. 2011; 4: 45–51.
93. Шабалина О.А. Модель процесса обучения и ее интерпретация в обучающей компьютерной игре // Вестник СГТУ. 2013. № 1 (70). URL: <https://cyberleninka.ru/article/n/model-protssessa-obucheniya-i-ee-interpretatsiya-v-obuchayushey-kompyuternoy-igre> (дата обращения: 12.05.2025).
94. Шабалина О.А., Воробкалов П.Н., Катаев А.В. Разработка обучающих игр: интеграция игровой и обучающей компоненты // Открытое образование. 2011. № 2-2. URL: <https://cyberleninka.ru/article/n/razrabotka-obuchayuschih-igr-integratsiya-igrovoy-i-obuchayushey-komponenty> (дата обращения: 12.05.2025).
95. Christopoulou E., Xinogalos S. Overview and comparative analysis of game engines for desktop and mobile devices // International Journal of Serious Games. 2017. № 4. С. 21–36. DOI: 10.17083/ijsg.v4i4.194.
96. Öztürk S. Analyzing maintainability factors in open-source game engines: implications for game developers // Multimed Tools Appl. 2025. DOI: 10.1007/s11042-025-20899-8.
97. Катаев А.В., Шабалина О.А. Открытая модель игрока для оценки знаний и навыков в компьютерных обучающих играх // Известия Волгоградского государственного технического университета. 2011. С. 79–85.
98. Shute V., Ke F., Wang L. Assessment and adaptation in games // DOI: 10.1007/978-3-319-39298-14.
99. Zhang Z., Huang X. Exploring the impact of the adaptive gamified assessment on learners in blended learning // Educ Inf Technol. 2024. DOI: 10.1007/s10639-024-12708-w.

100. Bastien J. Usability testing: a review of some methodological and technical aspects of the method // *Int J Med Inform.* 2009. № 79. e18–23. DOI: 10.1016/j.ijmedinf.2008.12.004.
101. Escudeiro P., Escudeiro N., Reis R., Barata A., Vieira R. Quality criteria for educational games // *EAI Endorsed Transactions on Serious Games.* 2013. № 1. DOI: 10.4108/trans.gbl.01-06.2013.e3.
102. Nas S., Yaman H. An evaluation of the educational games implementation process from the teacher candidates, students and parents' perspective // *Malaysian Online Journal of Educational Technology.* 2023. № 11. C. 291–308. DOI: 10.52380/mojet.2023.11.4.497.
103. Kiili K., Ketamo H., Kickmeier M. Evaluating the usefulness of eye tracking in game-based learning // *International Journal of Serious Games.* 2014. № 1. DOI: 10.17083/ijsg.v1i2.15.
104. Petri G., Gresse von Wangenheim C. A method for the evaluation of the quality of games for computing education. 2019.
105. Mousavi F., Norouzi Aval R., Sarbaz M., Kimiafar K. Evaluation tools for digital educational games: a systematic review // *Acta Medica Iranica.* 2022. № 8. C. 508–512. DOI: 10.18502/acta.v60i8.10835.
106. Vieira E., Silveira A., Martins R. Heuristic evaluation on usability of educational games: a systematic review // *Informatics in Education.* 2019. № 18. C. 427–442. DOI: 10.15388/infedu.2019.20.
107. Law L.-C., Sun X. Evaluating user experience of adaptive digital educational games with activity theory // *Int J Hum-Comput Stud.* 2012. T. 70. C. 478–497. DOI: 10.1016/j.ijhcs.2012.01.007.
108. Oliva-Zamora M., Larreina-Morales M. Cognitive accessibility in educational games: a set of recommendations // DOI: 10.1007/978-3-031-60049-4/6.
109. Bang H.J., Li L., Flynn K. Efficacy of an adaptive game-based math learning app to support personalized learning and improve early elementary school students' learning // *Early Childhood Educ J.* 2023. № 51. C. 717–732. DOI: 10.1007/s10643-022-01332-3.

110. Su F., Zou D. A systematic review of game-based assessment in education in the past decade // Knowledge Management & E-Learning. 2024. № 16(3). C. 451–476. DOI: 10.34105/j.kmel.2024.16.021.
111. Zapata-Rivera D. Adaptive, assessment-based educational games // 2010. № 6095. C. 435–437. DOI: 10.1007/978-3-642-13437-195.
112. Nyamsuren E., Vegt W., Westera W. Automated adaptation and assessment in serious games: a portable tool for supporting learning // DOI: 10.1007/978-3-319-71649-717.
113. Nielsen J. Usability Engineering. San Francisco: Morgan Kaufmann Publishers Inc., 1994.
114. Khalid M.S., Tretow T., Roark A. Usability evaluation of adaptive learning system Rhapsode™ Learner // Proceedings of the International Conference on Information and Communication Technology for Development. 2023. DOI: 10.1007/978-981-19-7528-86.
115. Ijsselsteijn W., De Kort Y., Poels K., Jurgelionis A., Bellotti F. Characterising and measuring user experiences in digital games // J Appl Mech. 2007.
116. Tretow T., Khalid M. S. Methods for evaluating learning analytics and learning analytics dashboards in adaptive learning platforms: a systematic review // Electron J E-Learn. 2023. № 21. C. 430–449. DOI: 10.34190/ejel.21.5.3088.
117. Hocine N. A systematic literature review of adaptive learning systems based on the assessment of collaboration quality // DOI: 10.5220/0013196300003932.
118. Liebenberg J., Klerck J. A tool for measuring educational game quality // DOI: 10.21125/inted.2023.1527.
119. Torrente J., Borro-Escribano B., Freire M., Blanco A., Marchiori E., Martinez-Ortiz I., Moreno Ger P., Fernández-Manjón B. Development of game-like simulations for procedural knowledge in healthcare education // IEEE Transactions on Learning Technologies. 2014. № 7. C. 69–82. DOI: 10.1109/TLT.2013.35.
120. Ak O. A game scale to evaluate educational computer games // Procedia - Social and Behavioral Sciences. 2012. № 46. C. 2477–2481. DOI: 10.1016/j.sbspro.2012.05.506.

121. Penha M., Correia W.F., Campos F., Barros M. Heuristic evaluation of usability — a case study with the learning management systems (LMS) of IFPE // International Journal of Humanities and Social Science. 2014. № 4. С. 295.
122. Magoulas G., Chen S., Papanikolaou K. Integrating layered and heuristic evaluation for adaptive learning environments // Электронный ресурс. 2003. DOI: 10.1016/j.sbspro.2012.05.506.

ПРИЛОЖЕНИЕ А. Пример построения пространства с пояснением фрагментов кода

Рассмотрим пример структуры курса в виде ориентированного графа (рисунок А.1).

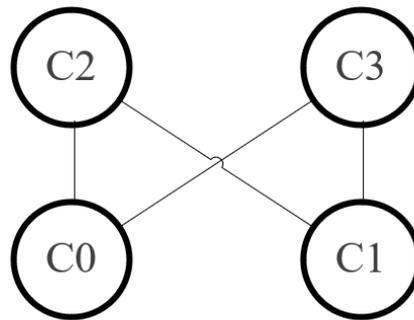


Рисунок А.1 – Граф

Входными данными для инструмента для построения пространства является матрица смежности, представленная в таблице А.1.

Таблица 3.1 – Матрица смежности

	1	2	3	4
1	0	0	1	1
2	0	0	1	1
3	0	0	0	0
4	0	0	0	0

В самом начале функции `graw`, принимаются значения в данную функцию `mm`-матрица смежности (двумерный массив из 1 и 0), x -позиции вершин по оси X , y позиции вершин по Y . Далее из массива `mm`, преобразуется в массив типа `numru` `m`, т.к. в дальнейшем метод `from_numru_matrix` принимает только массив `numru`. Строкой вида `G=nx.from_numru_matrix(m)` создается граф из матрицы смежности. Граф это, выражаясь языком объектно-ориентированного программирования,

экземпляр объекта Graph со своими методами и свойствами данного объекта из библиотеки networkx.

Для определения уровней по оси Y и соответствующих им вершин в графе осуществляется последовательный анализ данных (листинг А.1).

Листинг А.1 – Определение уровней по Y

```
ind_y=[]
for yy in y:
    if yy not in ind_y:
        ind_y.append(yy)
ind_y.sort()
for ind_y_ in ind_y:
    st_y_scet=0
    while j<len(y):
        if ind_y_==y[j]:
            st_y_scet+=1
        j+=1
    j=0
    st_y.append(st_y_scet)
```

На первом этапе производится поиск уникальных значений по оси Y, которые соответствуют различным уровням расположения элементов. Для этого создается пустой список ind_y, в который последовательно добавляются все уникальные значения yy из набора данных y, если они еще не присутствуют в списке. После этого список сортируется для упорядочивания уровней по возрастанию.

Далее, для каждого уровня по Y осуществляется подсчет количества вершин, расположенных на данном уровне. В цикле по индексам элементов данных происходит сравнение значения y[j] с текущим уровнем ind_y_. Если они совпадают, увеличивается счетчик st_y_scet, который фиксирует число вершин на данном уровне. После завершения внутреннего цикла значение st_y_scet добавляется в список st_y, что позволяет получить количество вершин на каждом уровне.

В результате анализа получаются уровни по Y : например, уровни 1, 1, 2, 2, а по оси X — значения 1, 2, 1, 2. Таким образом, выделяются два уровня по Y , на каждом из которых расположено по две вершины. Для построения обычного графа данной информации вполне достаточно, однако пространство знаний, представляемое решеткой, обязательно имеет точную верхнюю и точную нижнюю грани.

Имея позиции каждой вершины, количество уровней по Y и количество вершин на уровне, рассмотрим алгоритм построения точной нижней грани. Если на нижнем уровне одна вершина, значит добавочный элемент строить не нужно и данной элемент является точной нижней гранью, иначе в случае, когда на нижнем уровне более 1-ой вершины, создается добавочная нижняя вершина $C0$. А перед этим находим все вершины на нижнем уровне, циклично перебрав все значения позиций вершин Y у графа (листинг А.2). Если позиция соответствует самой нижней, значит вершина удовлетворяет условию и лежит в самом низу. После построения $C0$ строятся ребра для всех вершин, которые лежат внизу графа, т.е. соединяются с $C0$.

Листинг А.2 – До строение точной нижней грани

```

if st_y[0]>1:
    i=0
    akkum_x=0
    ver=[]
    ind=ind_y[0]
    while i<len(y):
        if y[i]==ind:
            ver.append(key[i])
            akkum_x+=x[i]
        i+=1
    G.add_node('C0')
    for v in ver:
        G.add_edge('C0',v)
    pos['C0']=np.array([akkum_x/(len(ver)), ind-2])

```

Для данного примера количество элементов на нижнем уровне равно $st_y[0] = 2$. В связи с этим необходимо дополнительно построить нижнюю границу, расположенную ниже текущего уровня (рисунок А.2).

Если количество элементов на уровне больше одного ($st_y[0] > 1$), то выполняется следующий алгоритм:

- инициализируются счетчики и переменные ($i=0$, $akkum_x=0$, $ver=[]$);
- определяется текущий уровень $ind=ind_y[0]$;
- в цикле по индексам элементов происходит сбор всех вершин, расположенных на данном уровне ($if\ y[i]==ind$), и суммирование их координат по оси X ($akkum_x+=x[i]$);
- после завершения цикла создается узел 'C0', который соединяется со всеми вершинами, расположенными на этом уровне;
- позиция узла 'C0' устанавливается как центр по оси X (среднее значение $akkum_x / len(ver)$) и смещается по оси Y вниз (на $ind-2$).

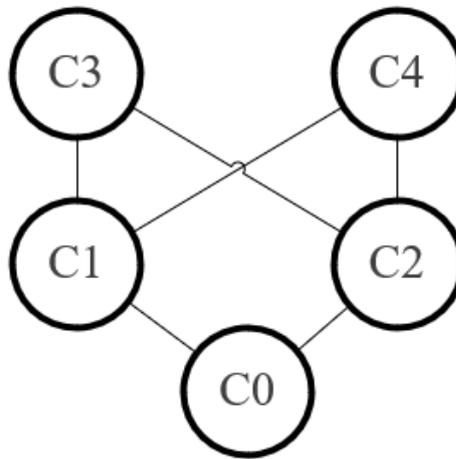


Рисунок А.2 – Граф с точной нижней гранью

Аналогично происходит построение точной верхней грани (листинг А.3). Циклично перебираются все вершины с позициями, если позиция соответствует последнему уровню по Y и вершина одна, тогда эта вершина является точной верхней гранью. Если вершин больше 1-ой, строится точная верхняя грань

уровнем выше. Далее происходит построение ребер из точной верхней грани к вершинам уровнем ниже. Для данного примера, на уровне $ind_y=2$ количество вершин $st_y=2$. В связи с этим необходимо дополнительно построить верхнюю границу уровня, расположенную выше текущего уровня (рисунок А.3).

Листинг А.3 – До строение точной верхней грани

```

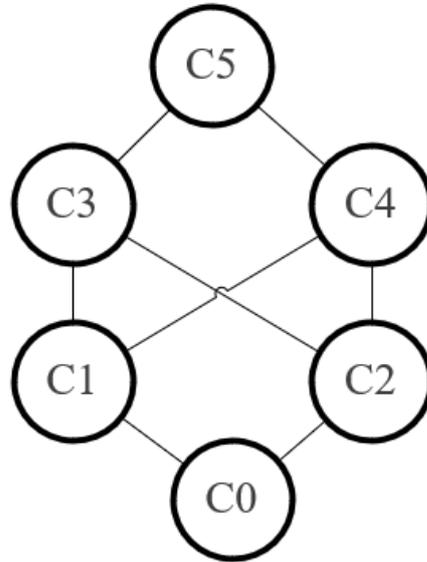
if st_y[len(st_y)-1]>1:
    i=0
    akkum_x=0
    ver=[]
    ind=ind_y[len(ind_y)-1]
    while i<len(y):
        if y[i]==ind:
            ver.append(key[i])
            akkum_x+=x[i]
        i+=1
    key_1=len(key)+1
    G.add_node(f'C{key_1}')
    for v in ver:
        G.add_edge(f'C{key_1}',v)
    pos[f'C{key_1}']=np.array([akkum_x/(len(ver)), ind+2])

```

Если количество вершин на последнем уровне $st_y[len(st_y)-1] > 1$, то выполняется следующий алгоритм:

- инициализируются счетчики и переменные ($i=0$, $akkum_x=0$, $ver=[]$);
- определяется текущий уровень $ind=ind_y[len(ind_y)-1]$;
- в цикле по индексам элементов происходит сбор всех вершин, расположенных на данном уровне ($if\ y[i]==ind$), и суммирование их координат по оси X ($akkum_x+=x[i]$);
- после завершения цикла создается новый узел с именем $f'C\{key_1\}'$, где key_1 — это следующий свободный номер узла ($len(key)+1$);
- этот узел соединяется со всеми вершинами, расположенными на текущем уровне;

- позиция нового узла устанавливается как центр по оси X ($\text{akkum_x} / \text{len}(\text{ver})$) и смещается вверх по оси Y на $\text{ind}+2$.



А.3 – Граф с точной нижней и точной верхней гранью

Помимо обязательных точной верхней и точной нижней граней, в пространстве знаний может присутствовать добавочный элемент (мнимая вершина), который образуется при смежности вершин двух уровней. Алгоритм построения мнимых вершин использует элементы комбинаторики, а именно выборка без повторений, и построен на поочередной проверке смежности вершин 2-х уровней. Сначала запускается цикл для всех уровней, если на уровне две или более вершин и на следующем по итерации уровне тоже две или более вершин, есть вероятность найти мнимую вершину.

Для данного примера, вершины двух уровней $\text{ind}_y = 1$ и $\text{ind}_y = 2$, являются смежными, поэтому добавляется мнимая вершина (рисунок А.4).

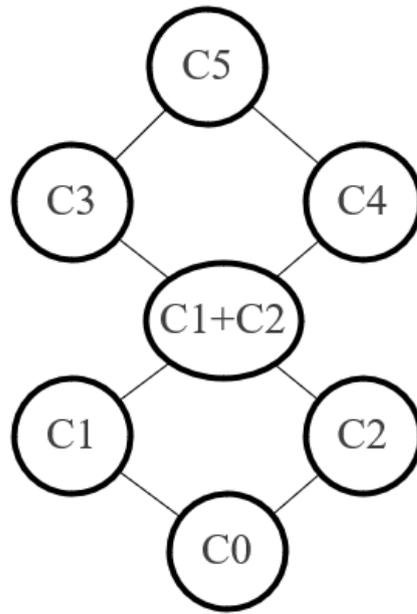


Рисунок А.4 – Пространство знаний

**ПРИЛОЖЕНИЕ Б. Копия свидетельства о государственной регистрации
программы ЭВМ**

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО
о государственной регистрации программы для ЭВМ
№ 2023618685

**Программа для поддержки разработки адаптивных
обучающих игр на Unity**

Правообладатель: *Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Волгоградский государственный технический
университет» (RU)*

Авторы: *Садовникова Наталья Петровна (RU), Шабалина
Ольга Аркадьевна (RU), Катаев Александр Вадимович
(RU), Хайров Александр Валерьевич (RU)*

Заявка № **2023617816**
Дата поступления **27 апреля 2023 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ **27 апреля 2023 г.**

*Руководитель Федеральной службы
по интеллектуальной собственности*



Ю.С. Зубов

ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ
Сертификат 68b80077c74c31013a94e6bd24145d5c7
Владелец **Зубов Юрий Сергеевич**
Действителен с 20.05.2022 по 26.05.2023

ПРИЛОЖЕНИЕ В. Копия акта о внедрении результатов исследования**ООО «К-Сервис»**

ИНН 3444004161 КПП 344401001

Адрес: 400087, г. Волгоград, ул. Невская, д. 7а;

тел: 8(8442)230000; e-mail: k-service@inbox.ru;

А К Т

об использовании результатов кандидатской диссертации
Хайрова Александра Валерьевича
на тему «Модели и методы автоматизации проектирования адаптивных
обучающих игр»

Настоящий акт составлен о том, что проектные и технологические решения, предложенные в диссертационной работе Хайрова А.В., используются в компании ООО «К-Сервис» для разработки адаптивного программного обеспечения (ПО) и отдельных программных модулей в составе программных продуктов.

Разработанные с применением модуля адаптации программные продукты востребованы на рынке ПО, так как обеспечивают персонафикацию процессов взаимодействия пользователя с ПО и повышают таким образом конкурентоспособность ПО.

Директор ООО «К-Сервис»



Факов Ю.А.