

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА ИГРЫ ДЛЯ САЙТА С ОБРАЗОВАТЕЛЬНЫМИ ИГРАМИ
ДЛЯ ДЕТЕЙ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Емелина Даниила Анатольевича

Научный руководитель
доцент, к. ф.-м. н.

А. С. Иванова

Заведующий кафедрой
к. ф.-м. н., доцент

С. В. Миронов

Саратов 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Основы разработки образовательных игр	5
1.1 Психолого-педагогические аспекты образовательных игр	5
1.1.1 Роль и значение игр в обучении детей	5
1.1.2 Психолого-педагогические подходы к разработке образовательных игр и их применение	5
1.2 Эффективные методы интеграции образовательного контента в игровые формы	6
1.3 Модели и методологии разработки игр	6
1.4 Инструменты разработки.....	7
1.4.1 Игровые движки.....	7
1.4.2 Графические библиотеки для разработки игр.....	8
1.4.3 Библиотеки для создания анимации	8
1.4.4 Прочие инструменты для создания игр	9
2 Разработка игрового приложения	10
2.1 Стартап как диплом.....	10
2.2 Описание используемых технологий	10
2.3 Star Races	11
2.3.1 Описание игры	11
2.4 Антураж игры.....	11
2.4.1 Управление.....	12
2.4.2 Правила игры.....	12
2.4.3 Интерфейс	13
2.4.4 Система уровней	14
2.4.5 Конец игры	14
2.4.6 Адаптивность интерфейса	14
ЗАКЛЮЧЕНИЕ	15

ВВЕДЕНИЕ

Актуальность темы: В современных условиях цифровизации образовательного процесса особое значение приобретает интеграция интерактивных технологий, таких как образовательные игры, в учебные программы для детей. Исследования показывают, что игровые формы обучения способствуют повышению мотивации, развитию когнитивных способностей и улучшению усвоения учебного материала у школьников.

Сайты с образовательными играми становятся все более популярными, так как они позволяют сочетать развлечение и обучение, предлагая детям интересные и познавательные задания. Разработка новой образовательной игры для такого сайта отвечает требованиям времени и способствует совершенствованию образовательного процесса, делая его более доступным и эффективным.

Кроме того, пандемия COVID-19 и связанный с ней переход на дистанционное обучение показали необходимость в качественных онлайн-ресурсах для детей. Традиционные методы обучения не всегда эффективны в удалённом формате, что вызывает потребность в инновационных решениях. Образовательные игры, разработанные с учётом возрастных особенностей и педагогических принципов, могут стать мощным инструментом для поддержания интереса и активности детей во время обучения.

Таким образом, актуальность данной работы заключается в создании ресурса, который не только развивает познавательные способности детей, но и адаптируется к новым условиям и вызовам современного мира.

Особое место в контексте данной работы занимает университетский проект «EduPlay», который разрабатывается командой студентов. Данный проект участвует в программе «**Стартап как диплом**» — инициативе, направленной на вовлечение талантливых студентов в развитие экосистемы технологического предпринимательства, а также на поддержку бизнеса, находящегося на начальной стадии.

Разработка и реализация программы обучения абитуриентов и преподавателей университетов в подготовке стартапов в качестве ВКР предусмотрены программой «Цифровая экономика».

Цель работы: Разработка интерактивной игры для сайта с образовательными играми для детей, способствующей улучшению усвоения учебного материала.

Задачи работы:

1. Рассмотреть психолого-педагогические аспекты образовательных игр.
2. Проанализировать основные методы интеграции образовательного контента в игровые формы.
3. Рассмотреть методы и инструменты разработки компьютерных игр.
4. Выбрать инструменты и технологии для разработки.
5. Разработать концепцию, сценарий, структуру и логику игры.
6. Создать дизайн интерфейса, учитывая возрастные особенности детей.
7. Реализовать основные игровые механики и функциональные возможности.

Структура и объем работы. Бакалаврская работа состоит из введения, 2 разделов, заключения, списка источников и 2 приложений. Общий объем работы — 85 страниц, из них 58 страниц — основное содержание, включая 41 рисунок, CD-диск в качестве приложения, список использованных источников — 22 наименований.

1 Основы разработки образовательных игр

1.1 Психолого-педагогические аспекты образовательных игр

1.1.1 Роль и значение игр в обучении детей

Игры играют важную роль в образовательном процессе детей, комбинируя обучение с развлечением, что повышает мотивацию и интерес к учебному материалу. В игровой форме дети легче усваивают сложные концепции, активируя познавательную активность и глубокое вовлечение. Игры позволяют экспериментировать, пробовать стратегии и получать мгновенную обратную связь, развивая критическое мышление и самостоятельность.

Кроме того, игры развивают широкий спектр навыков: решать проблемы, работать в команде, улучшать коммуникацию и творческое мышление. Они улучшают память, внимание и концентрацию, положительно влияя на успеваемость. Образовательные игры могут быть адаптированы под индивидуальные потребности, делая обучение более персонализированным и эффективным.

Игры также способствуют эмоциональному и социальному развитию детей. Элементы сотрудничества и соревнования учат управлять эмоциями, развивают эмпатию и социальные навыки. Игровая среда позволяет безопасно экспериментировать с социальными ролями, способствуя личностному росту и формированию устойчивых социальных навыков.

1.1.2 Психолого-педагогические подходы к разработке образовательных игр и их применение

Психолого-педагогические подходы к разработке образовательных игр основываются на понимании того, как дети учатся и взаимодействуют с информацией. Конструктивизм подчеркивает активную роль ученика, где обучение через опыт и экспериментирование играет ключевую роль. Образовательные игры создают интерактивные среды, стимулирующие познавательную активность и более глубокое понимание материала.

Теория множественных интеллектов Говарда Гарднера утверждает, что у каждого ребенка есть уникальный набор способностей. Игры должны предлагать разнообразные виды активности, затрагивающие разные типы интеллекта, чтобы каждый ребенок нашел свой способ взаимодействия и успешного усвоения материала.

Социокультурная теория Льва Выготского акцентирует значимость об-

щения и сотрудничества в обучении. Образовательные игры должны предусматривать условия для совместной деятельности и взаимодействия, улучшая когнитивные, социальные и коммуникативные навыки детей.

Таким образом, использование конструктивистского, когнитивного и социокультурного подходов при разработке образовательных игр способствует созданию продуктов, которые эффективно передают знания и способствуют всестороннему развитию детей.

1.2 Эффективные методы интеграции образовательного контента в игровые формы

При разработке игр для сайта с образовательными играми для детей эффективные методы интеграции образовательного контента могут повысить обучающую ценность и привлекательность игрового опыта. Один из методов — использование квестов и заданий, встроенных в игровую среду, что позволяет детям обучаться через выполнение игровых миссий.

Другой метод — использование историй и сюжетных линий, вовлекающих детей в игровой процесс и контекстуализирующих образовательный контент. Это делает обучение более увлекательным и запоминающимся.

Также эффективным является использование игровых элементов для проверки знаний: тесты, викторины и интерактивные задания позволяют детям проверять свои знания и получать обратную связь.

Наконец, персонализация и адаптация контента в играх улучшают процесс обучения, создавая более эффективную и индивидуализированную образовательную среду, учитывающую потребности и уровень знаний каждого ребенка.

1.3 Модели и методологии разработки игр

Разработка игр — сложный процесс, включающий множество методов. Основные из них:

Водопадная модель: последовательное выполнение этапов проекта (анализ, проектирование, реализация, тестирование, поддержка). Плюсы: управляемость и предсказуемость. Минусы: слабая гибкость к изменениям.

Scrum: гибкий метод, основанный на спринтах и самоуправляемых командах. Включает регулярные совещания и обратную связь, повышая эффективность и управляемость.

Kanban: метод управления, направленный на гибкость и прозрачность. Задачи визуально отслеживаются на доске Kanban, что позволяет эффективно распределять нагрузку.

Extreme Programming (XP): фокусируется на качестве кода и быстрой обратной связи. Включает парное программирование, тестирование итераций и постоянное обновление кода.

Lean: принцип устранения потерь и максимизации ценности. Ставится акцент на минимизации издержек и оптимизации процессов.

Каждая методология имеет свои особенности и полезна в разных контекстах. Водопадная модель полезна для проектов с четкими требованиями, Scrum и Kanban предлагают гибкость, XP акцентирует внимание на качестве кода, а Lean направлена на оптимизацию процессов. Эти методы позволяют создать качественные образовательные игры, соответствующие потребностям пользователей.

1.4 Инструменты разработки

Выбор правильных инструментов для разработки образовательных игр играет ключевую роль в создании качественного и увлекательного продукта для детей. Современные инструменты охватывают различные аспекты разработки, включая создание графики и анимации, написание и отладку кода, управление проектом и тестирование. Использование подходящих инструментов ускоряет процесс разработки и помогает создавать игры, удовлетворяющие образовательные потребности и интересы детей.

1.4.1 Игровые движки

Игровой движок представляет собой программную среду для разработки видеоигр. Он включает инструменты для рендеринга графики, физики, искусственного интеллекта, звука и анимации. Ранее разработчики создавали свои движки, что занимало много времени, но современные коммерческие движки упрощают и ускоряют процесс разработки игр.

Unity — мощный и гибкий движок для разработки 2D и 3D-игр, поддерживающий экспорт на множество платформ, включая WebGL. Он известен интуитивным интерфейсом и обширной документацией, что делает его доступным для разработчиков всех уровней.

Godot — бесплатный, кроссплатформенный и открытый игровой движок

для разработки 2D и 3D-игр. Он поддерживает экспорт на множество платформ, включая HTML5, и является отличным выбором для создания браузерных образовательных игр.

1.4.2 Графические библиотеки для разработки игр

Фреймворк — это структура для создания программного обеспечения, обычно связанная с определенным языком программирования. Фреймворки и графические библиотеки, такие как Phaser и Three.js, предоставляют инструменты для создания интерактивных игр, позволяя сосредоточиться на логике игры и дизайне.

Phaser — популярный HTML5-фреймворк для разработки 2D-игр на JS. Он прост в использовании и поддерживает анимацию, звук, физику и обработку ввода, что делает его подходящим для создания браузерных игр.

Three.js — библиотека JavaScript для создания 3D-графики в браузере с использованием WebGL. Она предлагает мощные инструменты для создания сложных 3D-сцен и анимаций.

PixiJS — графическая библиотека для создания 2D-игр, поддерживающая WebGL и Canvas. Она обеспечивает высокую производительность и простоту интеграции с другими библиотеками.

Babylon.js — библиотека для создания 3D-графики в браузерах. Она включает физический движок, систему частиц и звуковые эффекты, что позволяет создавать высококачественную 3D-графику.

Konva.js — библиотека для создания интерактивных HTML5-приложений с векторной графикой и анимацией. Она подходит для разработки как десктопных, так и мобильных приложений.

Выбор библиотеки или фреймворка зависит от конкретных задач и требований проекта. Phaser подходит для простых 2D-игр, тогда как Three.js и Babylon.js предлагают более продвинутые возможности для 3D-графики.

1.4.3 Библиотеки для создания анимации

Анимация — ключевой элемент в создании интерактивных интерфейсов, особенно в веб-разработке. JavaScript позволяет легко создавать динамичные визуальные эффекты благодаря библиотекам и фреймворкам.

Anime.js — легкая библиотека для создания 2D-анимаций с простым синтаксисом и высокими возможностями. Поддерживает CSS свойства, SVG, ани-

мацию пути. Недостатки: не подходит для 3D-анимаций и может не поддерживать старые браузеры.

GSAP — мощная библиотека для создания сложных анимаций с поддержкой CSS, SVG, Canvas, WebGL и числовых преобразований. Высокая производительность и обширная документация. Недостатки: сложный API, платные функции и возможные проблемы с интеграцией.

1.4.4 Прочие инструменты для создания игр

HTML — основной язык разметки для создания структуры веб-страниц. Легко интегрируется с CSS и JavaScript.

CSS — язык стилей для оформления веб-страниц. Позволяет создавать привлекательные и адаптивные дизайны.

Сборщики проектов — инструменты для автоматизации и оптимизации разработки. Включают минификацию, транспиляцию и управление зависимостями. Примеры: Webpack, Vite.

Пакетные менеджеры — инструменты для управления зависимостями и установки пакетов. Пример: npm для JavaScript и Node.js.

2 Разработка игрового приложения

2.1 Стартап как диплом

Игра разработанная в рамках данной бакалаврской работы предназначалась для университетского проекта «EduPlay», который участвует в программе «Стартап как диплом».

«Стартап как диплом» — это программа, которая направлена на вовлечение талантливых студентов в развитие экосистемы технологического предпринимательства, а также на поддержку бизнеса, находящегося на начальной стадии. Разработка и реализация программы обучения абитуриентов и преподавателей университетов в подготовке стартапов в качестве ВКР предусмотрены программой «Цифровая экономика».

EduPlay — это веб-портал, который разрабатывается командой студентов для обучения детей дошкольного возраста, учеников младших классов и родителей, желающих, чтобы обучение приносило их детям удовольствие

2.2 Описание используемых технологий

Современная веб-разработка основывается на HTML, CSS и JavaScript.

HTML определяет структуру веб-страниц, CSS отвечает за их визуальное оформление, а JavaScript добавляет интерактивность. Эти технологии необходимы для создания как статических сайтов, так и динамических веб-приложений.

Three.js — популярная библиотека JavaScript для 3D-графики, использующая WebGL. Она предоставляет инструменты для создания сложных 3D-сцен и обеспечивает высокую производительность. Преимущества Three.js включают удобный API, поддержку различных форматов 3D-моделей и активное сообщество. Библиотека оптимизирована для работы с большими объемами данных и поддерживает интеграцию с другими инструментами.

Vite — это инструмент для разработки веб-приложений, который обеспечивает быструю сборку и мгновенную перезагрузку модулей. Он использует ES Modules и поддерживает интеграцию с TypeScript и CSS-препроцессорами. Vite подходит для проектов с Three.js, так как ускоряет процесс разработки и улучшает производительность конечного продукта.

GSAP — библиотека для создания высокопроизводительных анимаций с гибким API. Она обеспечивает плавность анимаций и поддерживает анимацию свойств объектов Three.js. GSAP используется для создания динамичных веб-

приложений с анимациями, требующими высокой частоты кадров.

2.3 Star Races

2.3.1 Описание игры

Игра является раннером, где игрок управляет космическим кораблем, выбирая правильно написанные слова на русском языке. Этот жанр популярен на мобильных устройствах благодаря простой и захватывающей игровой механике, что позволяет играть на ходу. Игра способствует улучшению правописания слов у детей через интерактивное обучение и тренировки.

2.4 Антураж игры

Антураж игры включает в себя создание туннеля с помощью 3D-графики, который создает эффект движения и скорости для игрока. Туннель состоит из множества секций, каждая из которых содержит четыре стены: левую, правую, верхнюю и нижнюю. Для боковых стен используются случайные текстуры, что добавляет разнообразие в визуальный эффект. Туннель создается таким образом, чтобы создавать иллюзию бесконечности, перемещая секции в конец, когда они выходят из поля зрения игрока. Это поддерживает высокую производительность игры и минимизирует нагрузку на память компьютера.

Антураж игры также включает в себя создание футуристичной сетки красного цвета с помощью библиотеки Three.js. Эта сетка имеет пределы, количество делений и цвет линий, что создает высокотехнологичную игровую среду. Для анимации сетки определены переменные `moveableX` и `moveableZ`, позволяющие динамически перемещать линии и придавать игре живость.

Для визуальных эффектов и динамики используется `ShaderMaterial`, позволяющий гибко управлять шейдерами, переменными и временем. Вершинный шейдер определяет позицию линий, создавая иллюзию движения, а фрагментный шейдер настраивает цвет и прозрачность, делая сетку более выразительной и привлекательной визуально.

Таким образом, комбинация шейдеров вместе с технологией создания туннеля добавляет уникальный футуристичный стиль игре, обеспечивая высокую производительность и захватывающий игровой опыт.

Поле проверки в игре представляет собой прозрачный зеленый прямоугольник, созданный с использованием Three.js для проверки правильности выбранного ответа. Оно реализовано через `BoxGeometry` для задания размеров и

`MeshBasicMaterial` для задания цвета и прозрачности. Поле добавляется на сцену в реальном времени и обновляется после каждой проверки ответа.

Таким образом, добавление поля проверки вместе с футуристичной сеткой и созданием туннеля создает уникальный игровой мир с высокотехнологичным стилем, обеспечивая игрокам захватывающий и реалистичный игровой опыт.

2.4.1 Управление

Игра поддерживает управление кораблем как с клавиатуры, так и с сенсорного экрана, что обеспечивает доступность на различных платформах. Управление клавиатурой осуществляется через события `keydown` и `keyup`, а для сенсорного экрана — через `touchstart` и `touchend` для отслеживания движения пальцев. Эти механизмы позволяют игрокам интуитивно управлять кораблем, создавая плавное и реалистичное перемещение.

Функция `_animateObjectMove` управляет анимацией перемещения корабля, обеспечивая плавность и проверяя статус игры. При перемещении корабля происходит визуальное отображение его положения и активации карточек слов.

2.4.2 Правила игры

Игрок должен выбирать правильно написанные слова на русском языке, появляющиеся внизу экрана. У игрока три жизни: три неверных ответа приведут к окончанию игры. Счет игрока может быть бесконечным, что делает игру соревновательной. Лидерборды на сайте отображают конечные результаты игрока, мотивируя к соревнованию.

В игре есть несколько действий:

1. Выбор правильного слова добавляет 1 очко.
2. Выбор неверно написанного слова уменьшает количество жизней на 1.
3. Бездействие (не выбор слова) аналогично выбору неверного слова.

Игрок выбирает ответ, перемещая корабль над словом. Слова выбираются случайно и перемешиваются методом `_shuffleArray`. После формирования списка они отображаются в HTML структуре с помощью соответствующего метода.

2.4.3 Интерфейс

Интерфейс приложения разработан с использованием стандартного функционала HTML и CSS, включая семантические теги `<header>`, `<main>` и `<footer>`, что улучшает структурированность и доступность контента. CSS использован для стилизации элементов, адаптивной верстки и использования современных техник, таких как Flexbox.

Интерфейс игры разделен на три части: верхнюю, центральную и нижнюю. Верхняя часть содержит кнопку паузы для приостановки игры. Индикаторы жизней и счета отображаются в левом и правом верхних углах экрана соответственно.

Центральная часть включает меню паузы и меню конца игры, обеспечивая удобство и единообразие в интерфейсе. Меню паузы дает возможность управлять игровым процессом во время перерыва, а меню конца игры отображает достижения игрока и предлагает опции перезапуска или выхода из игры.

Нижняя часть интерфейса содержит карточки со словами, которые игрок должен анализировать и выбрать правильно написанное слово. Карточки реализованы как элементы списка `` в `<footer>` страницы.

Пример интерфейса представлен на изображении 2.1.



Рисунок 2.1 – Интерфейс игры

2.4.4 Система уровней

Игра включает несколько уровней, где меняется количество и разнообразие слов. На первом уровне игроку предлагается одно слово, на втором — несколько разных слов. Переход между уровнями осуществляется автоматически при достижении 5 очков. Вместе с увеличением уровня незначительно увеличивается время на определение правильного ответа, чтобы игрок мог справиться с разнообразием слов, сохраняя при этом сложность игры.

2.4.5 Конец игры

Игра, когда заканчиваются жизни игрока, показывает анимацию ломающегося и искрящегося корабля, а также отображает меню конца игры, убирая кнопку паузы. Для анимации падения корабля использовался метод `gsap.to()`, позволяющий контролировать параметры анимации, включая продолжительность, координаты, тип анимации и функции `onUpdate` и `onComplete`. Анимация поломки корабля реализована с использованием метода `_createExplosion` в `Three.js`, который генерирует взрыв из частиц в указанной позиции с настройкой количества частиц, цветов, размеров и скоростей.

2.4.6 Адаптивность интерфейса

Для повышения доступности игры на разных устройствах была внедрена адаптивность интерфейса и всей игры под различные разрешения экранов и соотношения сторон. Это позволяет пользователям комфортно играть на ПК, планшетах и мобильных телефонах. Адаптивность реализована через функции, которые изменяют размеры объектов при изменении размера экрана. Коэффициенты масштабирования для ширины и высоты вычисляются динамически и применяются к объектам, обеспечивая сохранение их визуальных свойств. Также реализован механизм сохранения положения корабля на экране при изменении разрешения с помощью параметра `Status`, который отслеживает его текущее состояние и позицию.

ЗАКЛЮЧЕНИЕ

В ходе выполнения бакалаврской работы были успешно решены поставленные задачи, что позволило создать качественный продукт, способный внести значительный вклад в сферу образовательных технологий. Рассмотрение психолого-педагогических аспектов образовательных игр помогло глубже понять потребности и особенности восприятия информации детьми разных возрастных групп, что легло в основу разработки интуитивного и эффективного интерфейса. Это знание стало фундаментом для создания увлекательного игрового процесса, способного стимулировать учебную мотивацию и улучшить усвоение знаний.

Изучение методов интеграции образовательного контента в игровые формы позволило определить наиболее эффективные подходы, способствующие гармоничному сочетанию обучения и игры. В ходе работы были исследованы различные модели и методологии разработки игр, что помогло выбрать оптимальные решения для реализации проекта. Разработка концепции и сценария игры основывалась на этих знаниях, обеспечивая содержательное и структурное наполнение игры, соответствующее образовательным целям.

Создание дизайна интерфейса, учитывающего возрастные особенности детей, стало важным этапом работы. Были выбраны и адаптированы инструменты и технологии, наиболее подходящие для разработки детских образовательных игр. Эти инструменты позволили создать визуально привлекательную и функционально удобную игру, способствующую легкому восприятию и взаимодействию. Разработанная архитектура и структура игры обеспечили логичное и последовательное развитие игрового процесса, поддерживая интерес и вовлеченность детей.

Реализация основных игровых механик и функциональных возможностей завершила процесс разработки, предоставив готовый продукт, готовый к использованию. Итоговая игра не только соответствует образовательным стандартам, но и предоставляет увлекательный способ обучения для детей, что делает ее ценным инструментом в образовательной среде. Таким образом, работа не только достигла поставленных целей, но и продемонстрировала потенциал использования игровых технологий для улучшения образовательного процесса.