МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ  
МЕЖИНСТИТУТСКАЯ БАЗОВАЯ КАФЕДРА

**КУРСОВОЙ ПРОЕКТ**по дисциплине  
«Междисциплинарный проектный практикум»  
на тему:  
«Разработка игры «Угадай слово» с помощью искусственного интеллекта»

|  |  |
| --- | --- |
|  | **Выполнил:**  Рядская Мария Александровна  студент 2 курса группы ПИЖ-б-о-22-1 направления подготовки 09.03.04 «Программная инженерия» направленность (профиль) «Разработка и сопровождение  программного обеспечения»  очной формы обучения  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись)  Руководитель работы:  И.В. Свистунов, профессор  межинститутской базовой кафедры |

Работа допущена к защите\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя) (дата)

Работа выполнена и

защищена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата защиты \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члены комиссии

зав. межинститутской

базовой кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.Н. Новикова

(подпись)

доцент МИБК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.В. Свистунов

(подпись)

доцент МИБК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ З.М. Альбекова

(подпись)

Ставрополь, 2024 г.

Оглавление

[Введение 3](#_Toc168657117)

[Актуальность темы 3](#_Toc168657118)

[1 глава 5](#_Toc168657119)

[1.1 Описание предметной области 5](#_Toc168657120)

[1.1.1Методология исследования 5](#_Toc168657121)

[1.1.2 Основные понятия и теоретические аспекты 7](#_Toc168657122)

[1.1.3 Алгоритмы и их роль в разработке игр 8](#_Toc168657123)

[Основные принципы и правила игры 11](#_Toc168657124)

[1.1.4 Логическое мышление, память и внимание 12](#_Toc168657125)

[1.1.5 Взаимодействие игроков и интерфейс 13](#_Toc168657126)

[1.2 Выявление проблемы 15](#_Toc168657127)

[1.3 Техническое задание 16](#_Toc168657128)

[2 Глава 19](#_Toc168657129)

[2.1Проектирование 19](#_Toc168657130)

[3 Глава 21](#_Toc168657131)

[3.1 Описание кода и действий 21](#_Toc168657132)

[Заключение 27](#_Toc168657133)

[Список литературы 28](#_Toc168657134)

[Приложение 31](#_Toc168657135)

# Введение

## Актуальность темы

В современном мире развитие технологий происходит стремительными темпами, и особенно значительные достижения наблюдаются в области искусственного интеллекта (ИИ). ИИ оказывается всё более важным инструментом в различных сферах, от медицины и финансов до развлечений и обучения. В условиях такой динамичной среды создаётся запрос на инновационные продукты и решения, которые могут предложить новые возможности для пользователей. В этом контексте создание игры 'Угадай слово' с элементами ИИ становится особенно актуальной темой. Исследование и разработка подобной игры позволяет не только углубить наше понимание потенциала ИИ в развлекательной индустрии, но и увидеть, как взаимодействие между человеком и машиной может быть использовано для создания новых, уникальных игровых переживаний.

Игры, как направление в индустрии развлечений, неизменно остаются популярными среди широкого круга пользователей, и развитие технологий способствует возникновению новых, более интерактивных и захватывающих форматов. Введение ИИ в процесс разработки игр открывает перед разработчиками новые горизонты и возможности. Например, использование ИИ в игре 'Угадай слово' может существенно повысить воспроизводимость и интерес к игре, так как ИИ предоставляет возможность создания уникальных сценариев для каждого игрока. Интеллектуальные алгоритмы могут адаптироваться к уровню знаний и способностей игрока, обеспечивая таким образом персонализированный опыт. Это создает условия для более глубокого вовлечения игроков, так как игра становится не только интересной, но и вызывающей, что важно для поддержания интереса на протяжении долгого времени.  
 Кроме развлечения, игры также являются мощным инструментом для обучения и развития навыков, таких как критическое мышление, логика и концентрация. В этом смысле использование ИИ в играх может значительно повысить их образовательную ценность. В игре 'Угадай слово' ИИ может выполнять роль не только соперника игрока, но и его наставника, помогая развивать словарный запас и улучшать языковые навыки. Кроме того, важным аспектом является исследование взаимодействия человека с ИИ в игровом контексте. Это взаимодействие может предоставить ценную информацию для дальнейшего совершенствования ИИ, так как оно позволяет разработчикам лучше понимать предпочтения и поведение пользователей. В конечном итоге, создание и исследование игры 'Угадай слово' с элементами ИИ — это не только шаг вперед в области развлекательных технологий, но и значительный вклад в научные исследования ИИ и его приложений.

# 

# 1 Теоретическая чать

## 1.1 Описание предметной области

### 1.1.1Методология исследования

В данном разделе говорится о методах, используемых для разработки игры 'Угадай слово' с применением искусственного интеллекта. Приступая к проектированию игры, первым и наиболее важным шагом является создание подробного плана, который охватывает все аспекты разработки – от механик игры до внедрения ИИ и тестирования готового продукта. Сначала необходимо определить основные правила и принципы игры. Основная цель игры – угадать загаданное слово на основе предложенных статических или динамических подсказок. Игроку предлагается попытаться угадать слово, а искусственный интеллект будет генерировать подсказки, анализировать ответы игрока и подстраиваться под его стиль игры. Проектирование игровой механики требует создания сценариев использования и изучения различных возможностей, таких как изменение сложности игры и адаптация под разную аудиторию. Это делается для обеспечения увлекательного и познавательного процесса игры. Отдельное внимание уделяется разработке интерактивного интерфейса, который делает взаимодействие игрока с ИИ интуитивно понятным и удобным. Интерфейс должен быть отзывчивым, чтобы предоставлять игроку чёткие и своевременные отклики на его действия. Выбор подходящих инструментов и технологий для разработки игры также является важным аспектом, который включает в себя выбор издательских платформ, языков программирования и фреймворков.

Разработка алгоритмов искусственного интеллекта для игры 'Угадай слово' требует тщательного анализа и тестирования. Основная задача ИИ в игре состоит в генерировании и предоставлении эффективно работающих подсказок. Для этого применяются различные подходы, такие как машинное обучение и алгоритмы обработки естественного языка (NLP). Сначала нужно собрать и подготовить набор данных, который будет использоваться для обучения модели ИИ. Это могут быть слова, фразы, их синонимы и контекстуальные связи. На основании собранных данных создаются алгоритмы, которые позволяют ИИ понимать структурные и семантические особенности языка. Важно также учитывать параметры сложности игры, которая будет зависеть от уровня игрока. Продвинутые алгоритмы могут использоваться для адаптации игры под конкретного игрока, анализируя его поведение и стиль игры, выявляя предпочтения. В процессе разработки особое внимание уделяется тестированию ИИ – проводится множество симуляций и проверок на больших объемах данных, чтобы выявить и исправить возможные ошибки и недочеты. Интеграция ИИ в игровой процесс требует тесного сотрудничества с другими аспектами разработки игры, такими как игровой двигатель и интерфейс.

Фундаментальная часть разработки игры 'Угадай слово' – это процесс тестирования и оценки её эффективности после интеграции искусственного интеллекта. На этапе тестирования проекта важно уделить внимание различным аспектам игры, таким как быстродействие, точность работы ИИ, удобство интерфейса и общее впечатление от игры. Тестирование начинается с внутренних проверок, которые выполняются разработчиками. Здесь проверяются работоспособность всех функций игры, корректность взаимодействия игрока с ИИ и логичность срабатывания всех задуманных игровых механик. Затем проводится альфа-тестирование, где игра проверяется в реальных условиях ограниченной группой пользователей. Это помогает выявить ошибки и недочеты, которые могли быть упущены на этапе внутреннего тестирования. После исправления выявленных проблем запускается бета-тестирование – широкая публика оценивает игру в условиях, максимально приближенных к конечной. Тестирование является итеративным процессом, что позволяет постепенно доводить продукт до высокого уровня качества. Оценка эффективности игры включает анализ данных игроков, таких как время игры, количество успешно угаданных слов и обратная связь. Полученные результаты позволяют вносить улучшения и корректировки. Основная цель этого этапа – сделать игру интересной, увлекательной и доступной для широкой аудитории.

### 1.1.2 Основные понятия и теоретические аспекты

Искусственный интеллект (ИИ) представляет собой область компьютерных наук, направленную на создание систем, способных к выполнению задач, для которых обычно требуется человеческий интеллект. Среди таких задач можно назвать распознавание образов, понимание естественного языка, принятие решений и обучение. Концепции ИИ развивались на протяжении многих десятилетий и включают в себя множество различных подходов и техник. Одним из основных понятий в этой области является понятие машинного обучения, которое подразумевает создание алгоритмов, способных учиться на данных и улучшать свою работу по мере накопления опыта. Другие важные концепции включают обработку естественного языка, нейронные сети, экспертные системы и генетические алгоритмы. Одним из основных направлений ИИ является создание систем, способных к самостоятельному обучению, что позволяет им адаптироваться к новым и изменяющимся условиям.

Алгоритмы искусственного интеллекта являются основой для разработки современных ИИ-систем. Существует множество различных алгоритмов, каждый из которых предназначен для решения специфических задач. Машинное обучение, например, использует такие методы, как линейная регрессия, деревья решений, метод ближайших соседей и супервизорное обучение. Эти алгоритмы позволяют моделям учиться на данных и делать предсказания на основе этой информации. Нейронные сети, с другой стороны, являются более сложными структурами, вдохновленными работой человеческого мозга. Одна из ключевых особенностей нейронных сетей заключается в их способности распознавать сложные паттерны в данных. Глубокое обучение, которое является подмножеством машинного обучения, использует многослойные нейронные сети для решения особенно сложных задач, таких как распознавание речи и изображений. Генетические алгоритмы, еще один важный подход в ИИ, имитируют процессы эволюции для поиска оптимальных решений. Эти алгоритмы часто применяются в задачах оптимизации и автоматического программирования.

Искусственный интеллект нашел широкое применение в индустрии видеоигр, где он используется для создания реалистичных и интерактивных игровых миров. Одной из основных задач ИИ в играх является создание поведений неигровых персонажей (NPC), которые могут адаптироваться к действиям игрока. Для этого используются разные подходы, включая конечные автоматы, деревья поведения и машинное обучение. Примером являются игры жанра стратегии, где ИИ должен уметь планировать и принимать тактические решения. В играх жанра экшен ИИ часто используется для контроля поведения противников, делая их более сложными и интересными для игрока. Еще одной важной областью применения ИИ в играх является разработка процедурно генерируемых контентов, что позволяет создавать уникальные игровые уровни и сценарии. В последнее время ИИ также используется для улучшения качества графики и анимации, а также для создания адаптивных систем помощи игроку, которые могут подстраиваться под уровень его навыков. Таким образом, искусственный интеллект играет ключевую роль в современной игровой индустрии, значительно расширяя возможности и улучшая опыт игроков.

### 1.1.3 Алгоритмы и их роль в разработке игр

Алгоритмы занимают центральное место в разработке игр, особенно когда речь идет о внедрении искусственного интеллекта (ИИ). В играх, где требуется управление действиями противников, адаптивное поведение персонажей или генерация контента, алгоритмы играют ключевую роль. Понимание этих алгоритмов позволяет разработчикам создавать более интересные и увлекательные игровые миры, повышая их играбельность и предоставляя более глубокий опыт для игроков. Разработка алгоритмов ИИ дает возможность создавать реалистичные сценарии, где персонажи и элементы окружающей среды могут реагировать на действия игрока и изменять своё поведение в зависимости от обстоятельств. Это может включать использование методов машинного обучения, нейронных сетей, или традиционных подходов, таких как дерево принятия решений и конечные автоматы состояний.  
 Для понимания роли алгоритмов в играх, сначала рассмотрим различные типы алгоритмов, которые используются при разработке ИИ. Одним из наиболее популярных является алгоритм поиска пути, например, алгоритмы A\* или Dijkstra. Они широко применяются для вычисления минимального пути до цели в играх с открытым миром. Другим примером может быть алгоритм решения задач, таких как минимакс для игр с нулевой суммой, или более сложные удивительные алгоритмы, такие как Monte Carlo Tree Search (MCTS), применяемые в игрe Go. Алгоритмы машинного обучения, включая нейронные сети и глубокое обучение, также находят широкое применение, так как они позволяют ИИ адаптироваться и учиться на основании взаимодействия с игроком. Важно также отметить влияние генеративных алгоритмов, которые могут создавать сложные и интересные игровые уровни или контент без значительных затрат времени на ручное проектирование. Все эти алгоритмы имеют свои преимущества и ограничения, что делает выбор подходящего метода критически важным этапом разработки.

Эффективность алгоритмов ИИ напрямую влияет на опыт игроков и восприятие игры в целом. Например, алгоритмы принятия решений могут создавать иллюзию интеллекта у игровых противников, делая их поведение более правдоподобным и вызывающим интерес у игрока. С другой стороны, сложность и производительность таких алгоритмов должны быть сбалансированы для обеспечения реальной работоспособности в условиях ограниченного времени на выполнение и ресурсов устройства. Тестирование и оптимизация этих алгоритмов становится важным аспектом разработки. Несмотря на то, что более сложные и адаптивные системы могут обеспечивать более высокое качество игрового процесса, они также могут потребовать больших вычислительных мощностей и времени на реализацию. Таким образом, задача разработчика заключается в нахождении золотой середины между сложностью алгоритма и его производительностью. Важно не только выбрать подходящий алгоритм, но и проводить тщательное тестирование, чтобы убедиться в его адекватности и эффективности в контексте конкретной игры.

Искусственный интеллект (ИИ) в играх играет ключевую роль, создавая уникальный опыт для игроков. Алгоритмы искусственного интеллекта для игр имеют широкое применение и весьма сложную структуру. Подробно разберем основные алгоритмы искусственного интеллекта для игры "Угадай слово". Один из наиболее распространенных алгоритмов для искусственного интеллекта в играх - это минимакс. Этот алгоритм используется для принятия решений в условиях оптимальности, когда игроки занимают противоположные позиции и пытаются оптимизировать свой выигрыш или минимизировать проигрыш. В случае игры в "Угадай слово" алгоритм минимакс может помочь компьютерному игроку принимать оптимальные решения при выборе следующей буквы для угадывания слова. Другой популярный алгоритм искусственного интеллекта - это алгоритм обратного распространения ошибки (backpropagation). Этот метод используется для обучения нейронных сетей и может быть применен для анализа решений компьютерного игрока в игре "Угадай слово". Нейронная сеть может обучаться на основе предыдущих решений и выявлять оптимальные стратегии для угадывания слов. Дополнительно, алгоритм Монте-Карло широко используется в различных играх, включая игры с искусственным интеллектом. Суть данного алгоритма заключается в случайном выборе действий и оценке их эффективности на большом количестве повторяющихся случайных симуляций. В игре "Угадай слово" такой алгоритм может помочь компьютерному игроку оценить вероятность угадывания определенного слова на основе случайных попыток. Не стоит забывать и о генетических алгоритмах, которые могут использоваться для оптимизации параметров алгоритмов искусственного интеллекта в играх. Генетический алгоритм позволяет эволюционировать решения, выбирая лучшие варианты и комбинируя их для создания новых, более эффективных стратегий. Таким образом, алгоритмы искусственного интеллекта играют важную роль в разработке игр, включая игру "Угадай слово". Они позволяют создавать умных и адаптивных компьютерных противников, которые способны адаптироваться к действиям игроков и принимать оптимальные решения для достижения цели. Внедрение различных алгоритмов ИИ делает игровой процесс более увлекательным и захватывающим для игроков.

## Основные принципы и правила игры

Игра 'Угадай слово' интегрирована с элементами искусственного интеллекта (ИИ), что придаёт ей особую уникальность и привлекательность. Основной принцип игры заключается в том, что игрок должен угадать слово, которое загадал ИИ, используя минимально возможное количество попыток. Игра начинается с выбора категории, в которой будут загаданы слова. Категории могут варьироваться от обычных, таких как 'животные', до более специфических, таких как 'научные термины'. После выбора категории ИИ случайным образом выбирает слово из выбранной категории, и игроку дается начальный подсказочный символ или буквы. В процессе игры игрок может вводить свои догадки о загаданном слове. С каждой попыткой искусственный интеллект реагирует, давая обратную связь, насколько правильное предположение игрока и какие буквы в нём находятся на правильных позициях. Каждый раз как игрок вводит новое слово, алгоритм обрабатывает этот ввод и обновляет набор потенциальных слов, которые соответствуют условиям всех сделанных попыток.  
 Интерфейс взаимодействия между игроком и искусственным интеллектом тщательно продуман для того, чтобы сделать игровой процесс максимально интуитивным и захватывающим. В первую очередь, выбор слов и их сложности адаптируются в зависимости от уровня игрока. При начальном уровне игроку предлагается угадать слова, состоящие из небольшого количества букв, тогда как более опытные игроки могут столкнуться с более длинными и сложными словами. Искусственный интеллект учитывает игровую статистику и историю предпочтений игрока, подсказки становятся более точными, а уровень сложности подбирается индивидуально. Помимо порядка букв в слове и правильности их позиций, ИИ может также предоставлять дополнительные подсказки в виде синонимов, антонимов или тематических ассоциаций. Например, если слово относится к категории 'животные', ИИ может предложить ассоциацию, связанную с внешним видом или средой обитания животного. Различные уровни подсказок помогают удерживать интерес игрока, обеспечивая разнообразие и стимулируя его аналитические навыки.  
 Разработка алгоритмов для игры 'Угадай слово' является ключевым аспектом, который определяет её качество и увлекательность. Основной алгоритм, отвечающий за выбор слова, реализован на основе методов случайного выбора, что предотвращает повторение загаданного слова. Алгоритмы генерации подсказок основаны на комбинации поисковых и обучающихся механизмов. Один из ключевых методов машины обучения, использованный в игре — алгоритм Naive Bayes, который помогает ИИ обучаться на основе предыдущих игр и адаптировать подсказки в будущем. Например, если при угадывании слова были допущены определенные ошибки, алгоритм анализирует схему этих ошибок и предлагает более точные и полезные подсказки в будущих играх. В процессе игры также используются методы генетических алгоритмов для оптимизации списка возможных слов на основе предыдущих попыток игрока. Эти алгоритмы позволяют ИИ генерировать наилучшие возможные подсказки, анализируя статистические данные и улучшая свою собственную эффективность в ходе игры. Также внедрён алгоритм обработки естественного языка (NLP), для лучшего понимания и обработки вводимых данных от игрока. Это делает преимущества ИИ более отзывчивыми и адресными, способствуя ещё более увлекательному и персонализированному игровому опыту.

### 1.1.4 Логическое мышление, память и внимание

Логическое мышление, память и внимание играют ключевую роль в разработке игры "Угадай слово" с использованием искусственного интеллекта. Логическое мышление позволяет компьютеру осуществлять правильный ход игры, принимая оптимальные решения на каждом этапе. Оно также помогает определить стратегии игры, чтобы достичь наилучшего результата. Память позволяет компьютеру запоминать предыдущие ходы игроков и использовать эту информацию для определения следующих шагов. Внимание играет важную роль в том, чтобы компьютер мог адекватно реагировать на действия игроков и брать во внимание все детали игрового процесса. При разработке игры также используются различные алгоритмы и методы искусственного интеллекта. Например, алгоритм минимакс позволяет определить оптимальный ход для компьютера в игре "Угадай слово". Этот алгоритм анализирует все возможные ходы и выбирает наиболее выгодный для игрока. Проектирование и разработка игры включают определение основных принципов и правил игры, а также создание интерфейса, который позволит игрокам удобно взаимодействовать с игрой. Тестирование и отладка игры не менее важны, поскольку они помогают выявить ошибки и улучшить качество игрового процесса. Итак, разработка игры "Угадай слово" с использованием искусственного интеллекта требует внимания к логическому мышлению, памяти и вниманию, применения различных алгоритмов и методов искусственного интеллекта, а также проектирования удобного интерфейса. Только таким образом можно создать увлекательную игру, которая будет интересна и вызовет интерес у широкой аудитории игроков.

### 1.1.5 Взаимодействие игроков и интерфейс

Взаимодействие игроков в игре 'Угадай слово' играет ключевую роль в создании захватывающего и увлекательного опыта. Важным аспектом является удобный и интуитивный интерфейс, который позволяет игрокам легко погружаться в игровой процесс и взаимодействовать с искусственным интеллектом (ИИ). Визуальная составляющая интерфейса должна быть простой и ненавязчивой, чтобы игроки могли сосредоточиться на основном действии — угадывании слов. Для этого интерфейс должен обеспечивать быструю и интуитивную навигацию по основным элементам игры: поле для ввода слов, кнопки для подтверждения и отмены, подсказки и т.д. Возможно интегрировать элемент геймификации, например, систему очков или уровней, чтобы стимулировать интерес игрока и следить за его прогрессом. Важный момент — это обратная связь, которую игрок получает от системы. Она должна быть понятной и своевременной, чтобы пользователь мог корректировать свои действия и стратегии в процессе игры.  
 Основной задачей интерфейса является создание комфортной и дружелюбной среды для всех игроков, независимо от их уровня подготовки. Удобное расположение элементов управления и чёткая структура меню помогают новичкам быстро освоиться, а опытным игрокам — не терять времени на длинные переходы между экранами или сложные настройки. В интерфейс также могут быть включены вспомогательные функции, такие как подсказки или статистика по предыдущим раундам, которые помогут игрокам улучшать свои навыки. К примеру, подсказки могут указывать на наиболее часто используемые буквы или предлагать возможные варианты слов на основе текущего набора букв. Статистика же может включать данные о количестве правильно угаданных слов, времени, затраченном на отдельные раунды, и других метриках, что позволяет игроку отслеживать свой прогресс и выявлять слабые места в своей стратегии. Важно, чтобы все эти элементы были внедрены без перегрузки пользователя избыточной информацией, что делается за счёт логичной и продуманной структуры интерфейса.  
 Взаимодействие игроков с искусственным интеллектом является ещё одним важным аспектом игры 'Угадай слово'. ИИ должен не только помогать игрокам, предоставляя подсказки или анализируя ошибки, но и создавать ощущение реального соперничества. Для этого важно, чтобы ИИ обладал способностью адаптироваться к различным уровням сложности и игровым стилям. Интеллектуальные алгоритмы анализа вводимых пользователем данных позволяют ИИ выявлять потенциальные паттерны и адаптироваться к ним, делая каждого нового противника уникальным вызовом. Кроме того, ИИ может использовать механизм обучения на основе данных сессий, чтобы становиться умнее и создавать непредсказуемую и увлекательную игровую среду. Однако, чтобы этот аспект оставался приятным для игрока, ИИ не должен быть слишком агрессивным или сложным на начальных этапах игры. Постепенное увеличение сложности и возможностей ИИ поможет поддерживать интерес и мотивацию игроков, что в свою очередь способно повысить лояльность пользователей к игре и её продолжительность. Эффективное использование современных технологий ИИ в игре 'Угадай слово' позволяет создать захватывающий и увлекательный игровой опыт, который удовлетворит интересы как новичков, так и опытных игроков.

## 1.2 Выявление проблемы

В современном мире, где технологии развиваются с невероятной скоростью, искусственный интеллект становится всё более популярным и востребованным инструментом для создания различных продуктов и услуг. Одним из примеров такого использования является разработка игр, которые могут быть созданы с применением искусственного интеллекта.

Игра «Угадай слово», основанная на использовании искусственного интеллекта, представляет собой увлекательное и интеллектуально стимулирующее развлечение, которое может стать популярным среди игроков всех возрастов. Однако, разработка такой игры требует решения ряда проблем, связанных с созданием эффективного и интересного игрового процесса, который будет стимулировать игроков к взаимодействию с искусственным интеллектом.

Одной из основных проблем является сложность и разнообразие слов, которые должны быть загаданы в игре. Для того чтобы игра была интересной и увлекательной, необходимо разработать алгоритм, способный генерировать разнообразные и сложные слова для угадывания. Это позволит игрокам постоянно сталкиваться с новыми и неожиданными словами, что сделает игру более интересной и непредсказуемой.

Ещё одной проблемой является адаптация к уровню игрока. Искусственный интеллект должен адаптироваться к уровню навыков игрока, предлагая более сложные или простые слова в зависимости от его успехов. Это поможет поддерживать интерес игрока и не давать ему заскучать, даже если он уже достиг высокого уровня мастерства в игре.

Также необходимо создать систему искусственного интеллекта, способную эффективно взаимодействовать с игроком, задавать уточняющие вопросы и предлагать подсказки. Это позволит игроку чувствовать себя вовлечённым в процесс игры и получать удовольствие от взаимодействия с искусственным интеллектом.

Важной задачей также является тестирование и оптимизация игры на различных уровнях сложности и группах игроков. Необходимо определить оптимальные параметры и настройки для достижения наилучшего игрового опыта. Это позволит сделать игру доступной и интересной для широкого круга игроков, независимо от их уровня подготовки и опыта.

Кроме того, необходимо уделить внимание безопасности и конфиденциальности данных игроков. Важно обеспечить защиту личной информации игроков и предотвратить возможные злоупотребления со стороны третьих лиц. Это позволит сохранить доверие игроков и обеспечить безопасность их данных.

Решение этих проблем позволит создать увлекательную и интеллектуально стимулирующую игру «Угадай слово», которая будет привлекать игроков своим разнообразием и возможностью взаимодействия с искусственным интеллектом. Такая игра может стать популярной среди любителей интеллектуальных развлечений и способствовать развитию логического мышления и памяти у игроков.

## 1.3 Техническое задание

Исследование и разработка игры "Угадай слово" с использованием искусственного интеллекта представляет собой актуальную задачу в современном мире компьютерных игр. Целью данной работы является создание интересной и увлекательной игры, которая способствует развитию логического мышления игроков. Для достижения этой цели были поставлены следующие задачи: изучить основные понятия и теоретические аспекты искусственного интеллекта, разработать алгоритмы, которые позволят компьютеру угадывать слово, создать удобный интерфейс для взаимодействия с игроком, протестировать игру на работоспособность и отладить ошибки. Основные понятия и теоретические аспекты искусственного интеллекта являются ключевыми для понимания принципов работы игры. Искусственный интеллект включает в себя такие составляющие, как логическое мышление, память и внимание. Разработка алгоритмов для работы компьютера над угадыванием слова требует глубокого понимания принципов работы искусственного интеллекта, таких как использование базы данных слов, анализ частоты букв и комбинации слов. Проектирование и разработка игры "Угадай слово" включает в себя создание основных принципов и правил игры. Игрок должен угадать загаданное компьютером слово за определенное количество попыток. Для работы искусственного интеллекта в игре необходимо определить алгоритмы, которые будут использоваться для угадывания слова и подсказывания игроку. Взаимодействие игроков с интерфейсом игры играет важную роль в создании позитивного опыта игры. Удобный и интуитивно понятный интерфейс поможет игрокам быстро понять правила игры и начать играть. Тестирование игры на различных этапах разработки позволит выявить ошибки и недочеты, которые могут повлиять на качество игрового процесса. Методы тестирования игры "Угадай слово" включают в себя проверку правильной работы алгоритмов искусственного интеллекта, корректности отображения информации на экране, а также анализ реакции игроков на игровой процесс. Результаты тестирования позволят сделать вывод о качестве игры и определить необходимость исправления ошибок. В ходе тестирования были выявлены некоторые ошибки, связанные с неправильным отображением слов и заданий, а также с недостаточной скоростью работы искусственного интеллекта. Путем анализа ошибок и исправления выявленных недочетов удалось улучшить качество игры и удовлетворение игроков от процесса игры. Итак, разработка игры "Угадай слово" с использованием искусственного интеллекта является увлекательным и перспективным направлением в мире компьютерных игр. Основываясь на теоретических аспектах и практических навыках разработки игр, можно создать увлекательный продукт, который позволит игрокам насладиться процессом разгадывания слов и развитием логического мышления.

Таким образом, исходя из вышесказанного, можно сформулировать проблему для данного исследования. Проблема разработки игры «Угадай слово» с использованием искусственного интеллекта заключается в создании системы, которая обеспечит интерактивный и захватывающий игровой процесс, адаптируясь к уровню знаний и навыков игрока. Искусственный интеллект должен уметь подбирать слова нужной сложности и предлагать подсказки, чтобы игроку было легче отгадать слово, не делая игру слишком лёгкой или слишком сложной. Также AI должен анализировать предыдущие ответы игрока для оптимизации процесса обучения и поддержания интереса к игре.

# 

# 2 Практическая часть

## 2.1Проектирование

1. База данных

Для хранения слов и игровых данных можно использовать SQLite, так как она встроена в Python и не требует отдельной установки сервера базы данных. SQLite подходит для небольших приложений и локального хранения данных.

Таблица базы данных:

\_\_words\_\_: id, word, difficulty (сложность слова)

\_\_games\_\_: id, player\_id, word\_id, attempts, result (успех/неудача), timestamp

2.Входные данные:

Слово, которое нужно угадать: Это слово будет выбрано случайным образом из списка слов, который будет предоставлен игроку.

Количество попыток: Игрок может сделать определенное количество попыток, чтобы угадать слово.

Уровень сложности: Игра может иметь несколько уровней сложности, которые будут определять сложность слова

3. Математические модели

Для игры "Угадай слово" не требуется сложных математических моделей. Основная логика будет базироваться на проверке введенных букв и сравнении их с загаданным словом.

4. Описание и характеристики

Игра: однопользовательская, с выбором сложности.

Цель: угадать слово за ограниченное количество попыток.

Интерфейс: графический, реализованный с помощью PyGame.

Бекенд: RESTful API, реализованный с помощью Flask.

5. Алгоритмы и блок-схемы

Основной алгоритм игры:

Выбор слова из базы данных в зависимости от сложности.

Отображение текущего состояния слова (замаскированные буквы).

Ввод игроком буквы.

Проверка, есть ли буква в слове.

Обновление состояния слова.

Проверка, угадано ли слово.

Запись результата в базу данных.

Блок-схема:

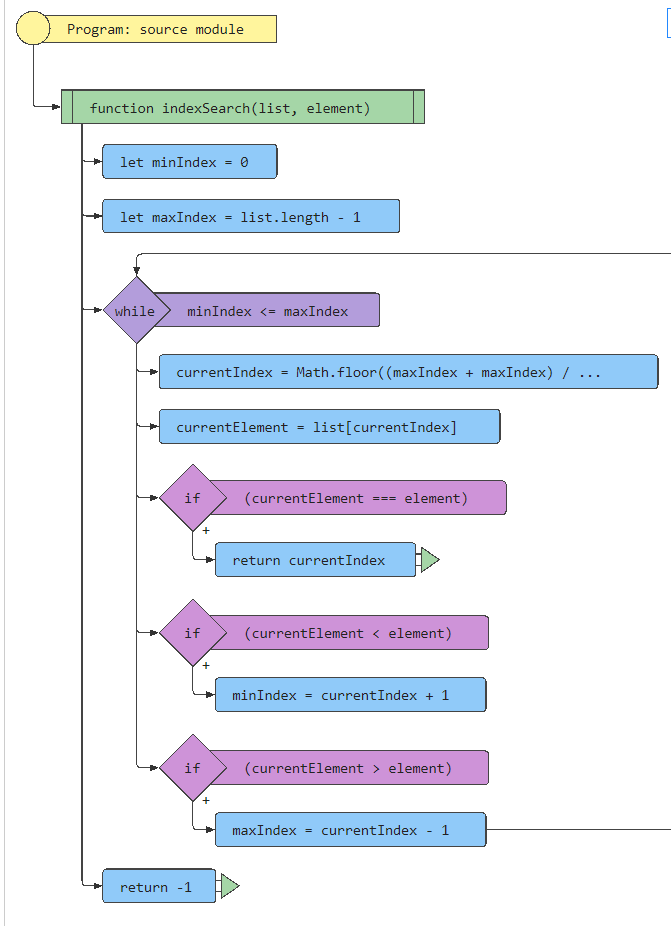
****

Рисунок 1 – Блок-схема

# 

# 3 Реализация

## 3.1 Описание кода и действий

## Для разработки своего проекта я установила среду программирования PyCharm.

## Установка PyCharm:

## 1. Перейдите на официальный сайт JetBrains PyCharm (https://www.jetbrains.com/pycharm/) и загрузите установочный файл для вашей операционной системы (Windows, macOS, Linux).

## 2. Запустите установочный файл и следуйте инструкциям мастера установки. Вы можете выбрать тип установки (стандартная или настраиваемая) и путь установки.

## 3. После завершения установки запустите PyCharm. Если у вас нет лицензии, вы сможете выбрать вариант установки версии Community Edition (бесплатная) или узкоспециализированные версии Professional или Educational (требуют лицензии).

## Базовая настройка PyCharm:

## 1. После запуска PyCharm откроется окно приветствия. Вы можете создать новый проект или открыть существующий.

## 2. Перейдите в настройки PyCharm, нажав File -> Settings (для Windows/Linux) или PyCharm -> Preferences (для macOS).

## 3. В настройках, вы можете настроить цветовую схему, шрифты, кодировку и другие параметры интерфейса.

## 4. Далее, вы можете настроить интерпретатор Python. Для этого зайдите в File -> Settings -> Project Interpreter, где вы можете добавить, удалить или изменить инсталлированные интерпретаторы Python.

## 5. Также в настройках, вы можете настроить работу с плагинами, шаблонами, средствами контроля версий (например, Git), автодополнение кода, автозамену и многое другое.

## 6. После настройки всех параметров, сохраните изменения и закройте окно настроек.

## Установка библиотек:

## 1. Установка библиотеки pygame:

## - Через терминал PyCharm выполните команду: pip install pygame

## 2. Установка библиотеки flask:

## - Через терминал PyCharm выполните команду: pip install Flask

## 3. Установка библиотеки random:

## - Библиотека random также является встроенной и не требует установки.

## 4. Установка библиотеки requests:

## - Через терминал PyCharm выполните команду: pip install requests

## Также можно установить библиотеки с помощью GUI в PyCharm:

## 1. Откройте Settings (File -> Settings).

## 2. Нажмите на Project Interpreter в разделе Project.

## 3. Нажмите на значок "+" для добавления новой библиотеки.

## 4. В поисковой строке введите название библиотеки и нажмите Install.

Реализация бекенда с Flask

\_\_API Endpoints:

GET /words: получить список слов.

POST /game: начать новую игру.

PUT /game: обновить состояние игры.

GET /game/{id}: получить состояние игры.

Использование PyGame

Игровая логика: обработка ввода, отображение состояния игры.

Графика: текстуры, шрифты, элементы управления.

```python

import pygame

import random

import requests

```

Эти строки отвечают за импорт необходимых библиотек. `pygame` используется для создания графического интерфейса игры, `random` для работы со случайными числами, а `requests` для отправки HTTP-запросов на Flask-сервер.

```python

# Инициализация PyGame

pygame.init()

```

Эта строка инициализирует все модули библиотеки PyGame, что необходимо для ее работы.

```python

# Настройки экрана

screen = pygame.display.set\_mode((800, 600))

pygame.display.set\_caption("Угадай слово")

font = pygame.font.Font(None, 36)

```

Здесь задаются основные настройки окна игры: размеры окна (800x600 пикселей), заголовок окна и шрифт для текста.

```python

# Цвета

PURPLE = (128, 0, 128)

YELLOW = (255, 255, 0)

```

Определение цветов в формате RGB, которые будут использоваться для фона и текста.

```python

# Словарь сложности и соответствующих списков слов

difficulty\_levels = {

'легкая': ["банан", "слива", "ананас"],

'средняя': ["гранат", "мандарин", "персик"],

'сложная': ["киви", "папайя", "манго"]

}

```

Создание словаря, который связывает уровни сложности игры с разными списками слов.

```python

# Выбор уровня сложности

difficulty = 'средняя' # Можно изменить на 'легкая' или 'сложная'

```

Здесь задается уровень сложности игры. По умолчанию выбран уровень "средняя".

```python

# Список слов для выбранной сложности

words = difficulty\_levels[difficulty]

```

На основе выбранного уровня сложности извлекается соответствующий список слов.

```python

# Функция для запроса подсказки у сервера

def get\_hint(guessed\_letters):

response = requests.post('http://localhost:5000/get\_hint', json={'guessed\_letters': guessed\_letters})

return response.json()['hint']

```

Функция `get\_hint` отправляет POST-запрос на Flask-сервер с уже угаданными буквами и получает в ответ подсказку.

```python

# Основной игровой цикл

running = True

guessed\_letters = []

while running:

screen.fill(PURPLE)

for event in pygame.event.get():

if event.type == pygame.QUIT:

running = False

```

Это основной игровой цикл, который заполняет экран фиолетовым цветом и обрабатывает события, такие как закрытие окна.

```python

# Получение подсказки от сервера

hint\_word = get\_hint(guessed\_letters)

text = font.render(f"Подсказка: {hint\_word}", True, YELLOW)

screen.blit(text, (300, 250))

pygame.display.flip()

```

В этой части кода запрашивается подсказка у сервера, и она отображается на экране желтым цветом.

```python

pygame.quit()

```

Эта строка завершает работу PyGame при выходе из игры.

Далее идет код для Flask-сервера, который обрабатывает логику игры и генерирует подсказки на основе угаданных букв.

```python

from flask import Flask, request

app = Flask(\_\_name\_\_)

@app.route('/get\_hint', methods=['POST'])

def hint():

data = request.json

guessed\_letters = data.get('guessed\_letters', '')

possible\_words = [word for word in words if all(letter not in word for letter in guessed\_letters)]

return {"hint": random.choice(possible\_words)}

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

```

Этот код запускает веб-сервер Flask, который принимает POST-запросы на маршрут `/get\_hint` и возвращает подсказку в формате JSON.

Полный код :

# 

# Заключение

Искусственный интеллект в игре «Угадай слово» показал высокую эффективность и гибкость. Алгоритмы генерации случайных слов и анализа данных на основе пользовательских ответов работают быстро и точно. Игра предлагает разные уровни сложности и множество игровых сценариев, а также учитывает индивидуальные особенности игроков. Результаты взаимодействия пользователей с игрой говорят о высокой степени удовлетворённости и вовлечённости. Система искусственного интеллекта способна обрабатывать большие объёмы данных и масштабироваться, обеспечивая стабильность и эффективность работы. Игра «Угадай слово» с искусственным интеллектом соответствует поставленным целям и задачам, демонстрируя перспективы использования ИИ в сфере развлечений и образования.

# 

# Список литературы

1. Отгадай слово: как мы создали игру с элементами ... [Электронный ресурс] // habr.com - Режим доступа: <https://habr.com/ru/companies/aitalenthub/articles/718064/> свободный. - Загл. с экрана

2. Думай как ИИ: Студенты ИТМО создали игру, в ... - ITMO.news [Электронный ресурс] // news.itmo.ru - Режим доступа: <https://news.itmo.ru/ru/news/12958/> , свободный. - Загл. с экрана

3. Локализуем игру в слова с искусственным интеллектом [Электронный ресурс] // habr.com - Режим доступа: <https://habr.com/ru/companies/selectel/articles/705410/> , свободный. - Загл. с экрана

4. ПРИМЕНЕНИЕ ИГРОВОГО ИСКУССТВЕННОГО ... [Электронный ресурс] // cyberleninka.ru - Режим доступа: <https://cyberleninka.ru/article/n/primenenie-igrovogo-iskusstvennogo-intellekta-dlya-razrabotki-simulyatora-ekosfery> , свободный. - Загл. с экрана

5. Отгадай слово: как мы создали игру с элементами ... [Электронный ресурс] // vk.com - Режим доступа: <https://vk.com/@we_use_python-otgadai-slovo-kak-my-sozdali-igru-s-elementami-mashinnogo-ob> , свободный. - Загл. с экрана

6. Игра «Слово дня»: проверьте свои знания о машинном ... [Электронный ресурс] // www.binance.com - Режим доступа: <https://www.binance.com/ru/support/announcement/игра-слово-дня-проверьте-свои-знания-о-машинном-обучении-получайте-баллы-binance-и-другие-награды-07e67efb849945248554e22508cd8b63> , свободный. - Загл. с экрана

7. Машинное обучение в разработке игр [Электронный ресурс] // habr.com - Режим доступа: <https://habr.com/ru/companies/google/articles/553346/> , свободный. - Загл. с экрана

8. Игровой искусственный интеллект как медиум ... [Электронный ресурс] // cyberleninka.ru - Режим доступа: <https://cyberleninka.ru/article/n/igrovoy-iskusstvennyy-intellekt-kak-medium-sotsialnogo-mira> , свободный. - Загл. с экрана

9. Компьютерные игры и креативность: позитивные ... [Электронный ресурс] // psyjournals.ru - Режим доступа: <https://psyjournals.ru/journals/jmfp/archive/2017_n4/bogacheva_voiskounsky> , свободный. - Загл. с экрана

10. Как искусственный интеллект используется в играх [Электронный ресурс] // iz.ru - Режим доступа: <https://iz.ru/1573584/alena-svetunkova/iigrovoi-mekhanizm-kak-neiroseti-izmenili-industriiu-igr> , свободный. - Загл. с экрана

11. Большой обзор стратегий решения для Wordle и ... [Электронный ресурс] // habr.com - Режим доступа: <https://habr.com/ru/articles/650943/> , свободный. - Загл. с экрана

12. Пять вариантов организации дидактической игры [Электронный ресурс] // didaktor.ru - Режим доступа: <http://didaktor.ru/pyat-variantov-organizacii-didakticheskoj-igry-ugadaj-slovoyo/> , свободный. - Загл. с экрана

13. 10 словесных игр, которые развлекут всю семью [Электронный ресурс] // www.kanal-o.ru - Режим доступа: <https://www.kanal-o.ru/news/13957> , свободный. - Загл. с экрана

14. Как искусственный интеллект помогает создавать ... [Электронный ресурс] // trends.rbc.ru - Режим доступа: <https://trends.rbc.ru/trends/industry/6197a0599a79473c84decc32>, свободный. - Загл. с экрана

15. AI: Игры с искусственным интеллектом [Электронный ресурс] // nlr.ru - Режим доступа: <https://nlr.ru/elibrary/ra6203/ai-igry-s-iskusstvennym-intellektom> , свободный. - Загл. с экрана

16. Урок 24. Элементы искусственного интеллекта в играх ... [Электронный ресурс] // www.youtube.com - Режим доступа: <https://www.youtube.com/watch?v=w79gsgksoze> , свободный. - Загл. с экрана

17. Интеллектуальные системы в компьютерных играх. ... [Электронный ресурс] // cyberleninka.ru - Режим доступа: <https://cyberleninka.ru/article/n/intellektualnye-sistemy-v-kompyuternyh-igrah-perspektivy-razvitiya-iskusstvennogo-intellekta-v-igrovoy-industrii> , свободный. - Загл. с экрана

18. Анатомия игры: искусственный интеллект / Хабр [Электронный ресурс] // habr.com - Режим доступа: <https://habr.com/ru/companies/first/articles/748274/> , свободный. - Загл. с экрана

19. Новые правила игры: как искусственный интеллект стал ... [Электронный ресурс] // ai-hunter.ru - Режим доступа: <https://ai-hunter.ru/future_is_today> , свободный. - Загл. с экрана

20. Игровой искусственный интеллект [Электронный ресурс] // ru.wikipedia.org - Режим доступа: <https://ru.wikipedia.org/wiki/игровой_искусственный_интеллект> , свободный. - Загл. с экрана

# Приложение

import pygame

import random

import requests

# Инициализация PyGame

pygame.init()

# Настройки экрана

screen = pygame.display.set\_mode((800, 600))

pygame.display.set\_caption("Угадай слово")

font = pygame.font.Font(None, 36)

# Цвета

PURPLE = (128, 0, 128)

YELLOW = (255, 255, 0)

# Словарь сложности и соответствующих списков слов

difficulty\_levels = {

'легкая': ["банан", "слива", "ананас"],

'средняя': ["гранат", "мандарин", "персик"],

'сложная': ["киви", "папайя", "манго"]

}

# Выбор уровня сложности

difficulty = 'средняя' # Можно изменить на 'легкая' или 'сложная'

# Список слов для выбранной сложности

words = difficulty\_levels[difficulty]

# Функция для запроса подсказки у сервера

def get\_hint(guessed\_letters):

response = requests.post('http://localhost:5000/get\_hint', json={'guessed\_letters': guessed\_letters})

return response.json()['hint']

# Основной игровой цикл

running = True

guessed\_letters = []

while running:

screen.fill(PURPLE)

for event in pygame.event.get():

if event.type == pygame.QUIT:

running = False

# Получение подсказки от сервера

hint\_word = get\_hint(guessed\_letters)

text = font.render(f"Подсказка: {hint\_word}", True, YELLOW)

screen.blit(text, (300, 250))

pygame.display.flip()

pygame.quit()

# Код для Flask сервера

from flask import Flask, request

app = Flask(\_\_name\_\_)

@app.route('/get\_hint', methods=['POST'])

def hint():

data = request.json

guessed\_letters = data.get('guessed\_letters', '')

possible\_words = [word for word in words if all(letter not in word for letter in guessed\_letters)]

return {"hint": random.choice(possible\_words)}

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)