

ОШ МАМЛЕКЕТТИК УНИВЕРСИТЕТИНИН ЖАРЧЫСЫ. ПЕДАГОГИКА. ПСИХОЛОГИЯ

ВЕСТНИК ОШСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА. ПЕДАГОГИКА. ПСИХОЛОГИЯ

JOURNAL OF OSH STATE UNIVERSITY. PEDAGOGY. PSYCHOLOGY

e-ISSN: 1694-8742

№1(6). 2025, 32-41

УДК: 378

DOI: [https://doi.org/10.52754/16948742_1\(6\)_4-2025](https://doi.org/10.52754/16948742_1(6)_4-2025)

ИГРОВОЙ СЕРВИС КАК ИНСТРУМЕНТ ДЛЯ СОВЕРШЕНСТВОВАНИЯ НАВЫКОВ ПРОГРАММИРОВАНИЯ

ОЮНДУН КЫЗМАТЫН КОЛДОНУУ ПРОГРАММАЛОО КӨНДҮМДӨРҮН ЖАКШЫРТУУ
КАРАЖАТЫ КАТАРЫ

GAME-BASED SERVICE AS A TOOL FOR ENHANCING PROGRAMMING SKILLS

Игнатьева Эмилия Анатольевна

Игнатьева Эмилия Анатольевна

Ignatieva Emilia Anatolyevna

кандидат психологических наук, доцент,

Чувашский государственный педагогический университет им. И. Я. Яковлева

психология илимдеринин кандидаты, доцент,

И. Я. Яковлев атындагы Чуваши мамлекеттик педагогикалык университети

Candidate Of Psychology Sciences, Associate Professor, I.Ya. Yakovlev Chuvash State Pedagogical University

iehmiliya@yandex.ru

ORCID: 0000-0002-4926-5371

ИГРОВОЙ СЕРВИС КАК ИНСТРУМЕНТ ДЛЯ СОВЕРШЕНСТВОВАНИЯ НАВЫКОВ ПРОГРАММИРОВАНИЯ

Аннотация

В работе рассматривается использование игровых элементов в обучении программированию для повышения мотивации студентов. Игровые сервисы представляют собой инновационный подход, который сочетает элементы геймификации и интерактивного обучения для создания увлекательного и эффективного образовательного опыта. Цель исследования — разработка и проверка обучающего сервиса GameProg, основанного на геймификации. Сервис предлагает пользователям решать задачи различной сложности, отслеживая их прогресс и предлагая достижения за выполнение задач. Эксперимент проведен среди студентов, обучающихся программированию, и показал повышение уровня заинтересованности на 25%. Большинство участников положительно оценили сервис, особенно отметив соревновательные элементы и систему прогрессии. В заключении подчеркивается, что внедрение игровых методов способствует эффективному совершенствованию навыков программирования, и такие элементы могут быть использованы для других образовательных целей. Полученные результаты демонстрируют перспективность геймификации в образовательном процессе.

Ключевые слова: геймификация, программирование, обучение, мотивация, игровой сервис.

Оюндуң қызметын колдонуу программалоо көндүмдөрүн жакшыртуу каражасы катары *Game-based service as a tool for enhancing programming skills*

Аннотация

Жумушта программалоого үйрөнүүдө оюндук элементтерди колдонуу студенттердин мотивациясын жогорулатуу максатында карапат. Оюндуң қызматтары – бул жаңычыл ыкма, ал геймификация жана интерактивдүү билим берүүнү бириктириүү аркылуу қызыктуу жана натыйжалуу билим берүү тажрыйбасын түзөт. Изилдеөнүн максаты – геймификацияга негизделген GameProg окуу қызметин иштеп чыгуу жана текшерүү. Қызмат колдонуучуларга ар кандай татаалдыгы бар тапшырмаларды чечүүнү сунуштайт, алардын жетишкендиктерин көзөмөлдөп, тапшырмаларды аткаруу үчүн жетишкендиктерди берет. Эксперимент программалоону үйрөнүп жаткан студенттер арасында жүргүзүлүп, қызыгуунун деңгээли 25% га жогорулоо көрсөтүлгөн. Кошумча катышуучулардын көпчүлүгү қызматты оң баалашып, атайын атаандаштык элементтерин жана прогресс системасын өзгөчө белгилешти. Түшүнүктө оюн ыкмаларын киргизүү программалоо көндүмдөрүн натыйжалуу жакшыртууга көмөктөшөрүн жана мынтай элементтер башка билим берүү максаттары үчүн дагы колдонулушу мүмкүн экендиги баса белгиленет. Алынган жыйынтыктар геймификациянын билим берүү процессинде чон мүмкүнчүлүктөргө жетишүүгө көрсөтүп турат.

Ачыкч сөздөр: геймификация, программалоо, билим берүү, мотивация, оюндук қызмет.

Abstract

This study explores the use of game-based elements in programming education to enhance student motivation. Game services represent an innovative approach that combines gamification and interactive learning to create an engaging and effective educational experience. The aim of the research is to develop and test the GameProg learning platform based on gamification principles. The service offers users tasks of varying difficulty, tracks their progress, and rewards achievements for completed tasks. An experiment conducted among programming students showed a 25% increase in engagement levels. Most participants responded positively to the platform, particularly highlighting the competitive elements and the progression system. The study concludes that integrating game-based methods contributes to the effective improvement of programming skills and that such elements can be applied to other educational goals as well. The results demonstrate the potential of gamification in the educational process.

Keywords: gamification, programming, learning, motivation, game-based service.

Постановка проблемы, ее актуальность и научная новизна

В условиях стремительного развития информационных технологий и постоянного обновления требований к профессиональным навыкам выпускника программирование становится одной из ключевых компетенций, востребованных на рынке труда. Однако, традиционные методы обучения программированию зачастую не обеспечивают достаточного уровня мотивации и вовлеченности учащихся, что приводит к снижению эффективности освоения материала. В результате преподаватели и исследователи изучают новые подходы, позволяющие сделать обучение программированию более увлекательным и приятным. В связи с этим возрастающий интерес к геймификации и интерактивным методам обучения, основанным на игровых механиках, становится важным направлением педагогических исследований.

Проблема, рассматриваемая в этом исследовании, заключается в том, как эффективно повысить мотивацию и интерес учащихся к обучению программированию с помощью инновационных методов. Использование геймификации, которая интегрирует игровые элементы в среду обучения, представляет собой потенциальное решение. Делая процесс обучения более интерактивным, соревновательным и полезным, геймификация может помочь повысить общую вовлеченность учащихся и результаты обучения. Однако необходимы эмпирические данные, подтверждающие эффективность такого подхода в контексте обучения программированию. Научная новизна исследования заключается в разработке игрового обучающего сервиса для программирования, который включает такие элементы, как система прогрессии, достижения и соревновательные механизмы. Исследование направлено на оценку эффективности использования этих игровых элементов для улучшения учебных результатов.

Анализ поставленной проблемы

Проблема, поставленная в исследовании, заключается в поддержании мотивации учащихся к обучению программированию, что имеет решающее значение для эффективного обучения и развития навыков. Программирование по своей природе требует как теоретического понимания, так и значительной практической работы, что делает его предметом, который может легко оказаться непосильным для учащихся, особенно начинающих. Традиционные педагогические методы часто в значительной степени основаны на этих абстрактных понятиях и повторяющихся упражнениях по кодированию, что со временем может привести к снижению вовлеченности. Следовательно, такая отстраненность препятствует прогрессу учащихся и развитию критического мышления и навыков решения проблем, необходимых для программирования.

Геймификация, как методология, активно исследуется в образовательной среде с целью повышения мотивации и вовлеченности учащихся. Применение игровых элементов в неигровых контекстах было впервые рассмотрено Детердингом С. и его коллегами, и с тех пор этот подход начал активно развиваться как в зарубежных, так и в российских исследованиях (Детердинг, 2011). И. Е. Гусев подчеркивает, что игровые элементы в обучении способствуют развитию познавательных интересов и мотивации учащихся, особенно в тех дисциплинах, где требуется абстрактное мышление, например в программировании (Гусев, 2023). Н. Ф. Талызина, классик отечественной педагогики, исследовала вопрос программированного обучения, что является одним из прототипов современных адаптивных и игровых подходов. Она указывала на важность использования заранее спланированных шагов и контрольных точек, что перекликается с современными системами достижения и уровней в геймификации. (Талызина, 1969). Кроме того, С. И. Горлов проводил исследования по интеграции игровых

методов в образовательные платформы. Он акцентирует внимание на том, что игровые элементы, такие как уровни, рейтинги и баллы, позволяют студентам не только соревноваться между собой, но и следить за собственным прогрессом, что стимулирует более активное участие в учебном процессе. При рассмотрении ресурсов геймификации в образовании (Горлов, 2023). А.С. Ветушинский дал определение явлению геймификации как метод работы с поведением человека на основе использования элементов и средств игрового мышления (Ветушинский, 2020). Взаимодействие в рамках геймифицированного обучения позволяет создать среду, в которой студенты могут активно взаимодействовать с образовательным контентом и друг с другом (Игнатьева & Никитин, 2022). Взаимодействие в реальном времени через игровые платформы и мобильные приложения обеспечивает мгновенную обратную связь, что помогает студентам быстро исправлять свои ошибки и развивать навыки. В программировании, где учащиеся часто сталкиваются с трудностями из-за сложных и абстрактных концепций, геймификация может облегчить процесс освоения материала. Однако не все существующие платформы предоставляют адекватный баланс между игровыми элементами и образовательными целями.

Предложения автора по решению проблемы

Методологическую основу исследования проблем программированного обучения составляют бихевиористическая теория учения (в зарубежной дидактике), деятельностная теория научения и основные идеи кибернетики (отечественная дидактика). Рассмотрим сущность обозначенных подходов. В целом сущность программированного обучения состоит в следующем: учебный материал разделяется на небольшие логически завершенные дозы (части), легко усваиваемые учащимися (эти дозы (части) получили название «шаги», «порции», «фрагменты информации»); учебный процесс состоит из последовательных тесно взаимосвязанных между собой шагов, содержащих порцию знаний и мыслительных действий по их усвоению; каждый шаг завершается контролем (вопросом, заданием и т. д.); при правильном выполнении контрольных заданий (выборе правильного ответа) ученик получает новую порцию информации; при неправильном ответе ученик получает разъяснение и указание вновь выбрать правильный ответ; каждый учащийся работает самостоятельно и овладевает учебным материалом в оптимальном для него темпе; результаты выполнения всех контрольных заданий фиксируются и становятся известны как самому ученику, так и учителю. Это создает для ученика возможность самоконтроля (внутренняя обратная связь), а для учителя возможность получения информации о ходе познавательной деятельности каждого учащегося (внешняя обратная связь); педагог выступает организатором обучения и помощником (консультантом) в случае затруднений учащихся при работе над «шагом», «дозой» информации, осуществляет индивидуальный подход; в учебном процессе широко применяются специфические средства: программируемые учебники, программируемые учебные пособия, обучающие машины.

Педагогическая оценка программированного обучения. Достоинством этого вида обучения является получение субъектами образовательного процесса (учителем и каждым учеником) полной и постоянной информации о степени и качестве усвоения всей учебной информации. Это обеспечивается установлением прочной внешней и внутренней обратной связи, лежащей в основе функционирования системы «учитель-ученик» как целостной системы, в которой обучающие усилия учителя (компьютера) совпадают с собственными усилиями школьника. Поэтому обеспечивается высокий результат. Еще одно преимущество состоит в том, что в этой системе ученик занимает позицию не только объекта, но и позицию

субъекта собственной учебной деятельности.

Проблемно-ориентированное обучение (Problem-Based Learning, PBL). Концепция предлагает студентам решать реальные проблемы или задачи, которые требуют применения программирования. Задачи могут быть связаны с разработкой приложений, созданием веб-сайтов, решением алгоритмических задач и т. д. Основная идея состоит в том, чтобы студенты активно применяли свои знания программирования для решения конкретных проблем.

Концепция проектного обучения предлагает студентам работать над длительными проектами, которые требуют разработки программного кода. Проекты могут быть индивидуальными или групповыми и могут включать различные этапы, такие как анализ требований, проектирование, разработка и тестирование. В рамках данной концепции студенты разрабатывают навыки программирования, решая реальные задачи.

Игровые подходы к обучению (Gamification). Игровые элементы и принципы могут быть интегрированы в обучающий процесс, чтобы повысить мотивацию и вовлеченность студентов. Например, можно использовать геймификацию для создания конкурентной среды, где студенты могут соревноваться между собой или получать награды за достижения в программировании. Игровые элементы, такие как баллы, достижения, уровни и рейтинги, могут стимулировать студентов к активному изучению и практике программирования.

Адаптивное обучение. Концепция предлагает индивидуализированный подход к обучению, учитывая индивидуальные потребности и способности студентов. В рамках обучения программированию это может означать предоставление студентам учебных материалов и заданий, соответствующих их уровню знаний и навыков. Адаптивное обучение может включать в себя использование онлайн-платформ с интерактивными заданиями и автоматической обратной связью.

Алгоритмизированное обучение строится на основе разработки соответствующих моделей мыслительных процессов, последовательных умственных действий, обеспечивающих решение учебных задач. Алгоритмизация обучения состоит в разработке и реализации алгоритмов для учащихся или алгоритмов для обучающих лиц (или машин). Важной теоретико-методологической основой этого вида обучения, как и программированного обучения, является кибернетический подход. Основная цель алгоритмизированного обучения — повышение эффективности управления процессом обучения. Деятельность учителя по алгоритмизации деятельности учащихся, то есть разделение ее на отдельные взаимосвязанные элементы (действия, шаги), состоит из операций: выделение условий, необходимых для осуществления обучающих действий; выделение самих обучающих действий; определение способов связи обучающих и учебных действий.

В таблице представлены рассмотренные платформы игрового обучения программированию в срезе представленных в них функциональных возможностей.

Таблица. Анализ платформ игрового обучения программированию

Список характеристик	Codewars	CodinGame	LeetCode
Выбор из нескольких языков	+	+	+
Градация заданий по уровню сложности	+	+	+
Градация заданий по затрагиваемым темам	+	-	-
Наличие справочной информации	-	+	+
Подготовка к интервью	-	-	+
Система уровней пользователей	+	+	-
Соревновательные элементы	+	+	+
Система достижений	+	+	-
Наличие платной подписки	+	-	+
Наличие возможности ознакомиться с решениями других людей	+	+	+

На основе рассмотренных игровых сервисов Codewars, CodinGame и Leetcode, был выделен следующий *перечень функциональных возможностей необходимых для построения игрового обучающего сервиса*:

- возможность создания задач под разные языки программирования;
- возможность разделения задач по уровню сложности задачи, требуемому навыку программиста;
- функциональная возможность, выделять к какой именно концепции программирования относится каждая отдельно взятая задача;
- наличие автоматизированной среды тестирования пользовательских решений;
- наличие прогрессии пользователя по мере решения им задач;
- наличие системы достижений, основанной на пользовательском прогрессе;
- возможность пользователю сравнивать собственную прогрессию на основе прогрессии других пользователей игрового сервиса, через функционал топа игроков;
- возможность рассмотрения решений других игроков к уже решенным задачам;
- наличие администраторского интерфейса с соответствующим функционалом, обеспечивающим администрирование базы курсов, заданий, тем заданий, достижений и рядовых пользователей системы;

К входным данным системы относятся:

- перечень поддерживаемых системой языков программирования, каждый из которых представлен в виде отдельного курса;
- основная информация по каждому рядовому пользователю информационной системы;
- перечень поддерживаемых концепций программирования;
- основная информация по каждой задаче, представленной на сервисе, в том числе:
 - язык программирования задачи;
 - наименование задания;
 - необходимые для решения концепции программирования;
 - текстовое описание задачи;
 - перечень тестовых условий выполнения задачи для автоматизированной системы тестирования;
 - уровень сложности задачи, определяющий итоговую ценность предоставленного решения, в опытном эквиваленте;
 - основная информация по каждому представленному на сервисе достижению, в том числе:
 - наименование достижения;
 - текстовое описание достижения, с условием его получения;
 - перечень условий получения достижения пользователем для автоматизированной системы оценки прогресса пользователя;
 - уровень сложности достижения, определяющий итоговую ценность полученного пользователем достижения, в опытном эквиваленте.

К выходным данным системы относятся:

- перечень имеющихся в базе решений по каждой задаче, каждого решившего эту задачу пользователя;
- рассчитывающийся в автоматическом режиме прогресс пользователя, основанный на предоставленных им решениях, в том числе:

- количество набранного опыта, являющегося совокупной суммой всего опыта, полученного пользователем за предоставленные решения и полученные достижения, в соответствии с их уровнем сложности;
- текущий уровень пользователя, основанный на набранном им опытом;
- необходимое количество опыта, до следующего уровня;
- полный перечень полученных пользователем достижений;
- полный перечень еще не полученных пользователем достижений с условием их получения.
- топ имеющихся в системе пользователей, место в котором определяется на основе совокупного, набранного каждым пользователем опыта, полученного за предоставленные решения и полученные достижения в соответствии с их уровнем сложности.

В информационной системе должно присутствовать разграничение прав редактирования данных. Правом на ввод начальных данных обладает. Пользователи могут записываться на курсы, решать выбранные задачи, просматривать собственный прогресс.

Для решения проблемы адаптации игровых подходов к различным уровням знаний студентами факультета физико-математического образования информатики и технологий, был разработан игровой сервис для изучения языков программирования, который объединяет игровую мотивацию с образовательными целями.

Основные элементы сервиса включают:

1. Градацию сложности — задачи различной сложности, разделенные по уровням, чтобы обеспечить постепенное повышение навыков программирования.
2. Система достижений и прогрессии — пользователи могут отслеживать свои успехи, получая очки и достижения за выполнение задач.
3. Соревновательные механизмы — топ игроков, который стимулирует студентов к повышению своей активности.
4. Автоматизированная проверка решений — позволяет моментально получать обратную связь, что повышает эффективность обучения.

Геймификация нами рассматривается как процесс привнесения игровых элементов в изначально неигровой вид деятельности, с целью повышения мотивации и заинтересованности. Разработка игрового сервиса была осуществлена на языке программирования Python, для реализации базы данных была использована, встроенная в Python, СУБД SQLite. Веб составляющая системы реализована с помощью свободного фреймворка Django. На основе анализа предметной области была сформирована структура модели базы данных игрового обучающего сервиса GameProg (рис.).

В его основе лежат принципы программирования с элементами геймификации. Сервис предлагает пользователям решать задачи различной сложности, отслеживая их прогресс и предлагая достижения за выполнение задач. Основные функции сервиса включают автоматизированную проверку решений, систему прогрессии и топ пользователей для создания соревновательной среды.

На основе анализа предметной области была сформирована структура модели базы данных игрового обучающего сервиса GameProg состоящая из следующих таблиц (рисунок):

- «Пользователи» - одна из стандартных таблиц в ORM Django, содержащая всю необходимую информацию по зарегистрированным в системе пользователям. Каждое поле обладает следующими атрибутами: идентификатор пользователя (первичный ключ), пароль, дата последней авторизации, наличие администраторских прав, логин;

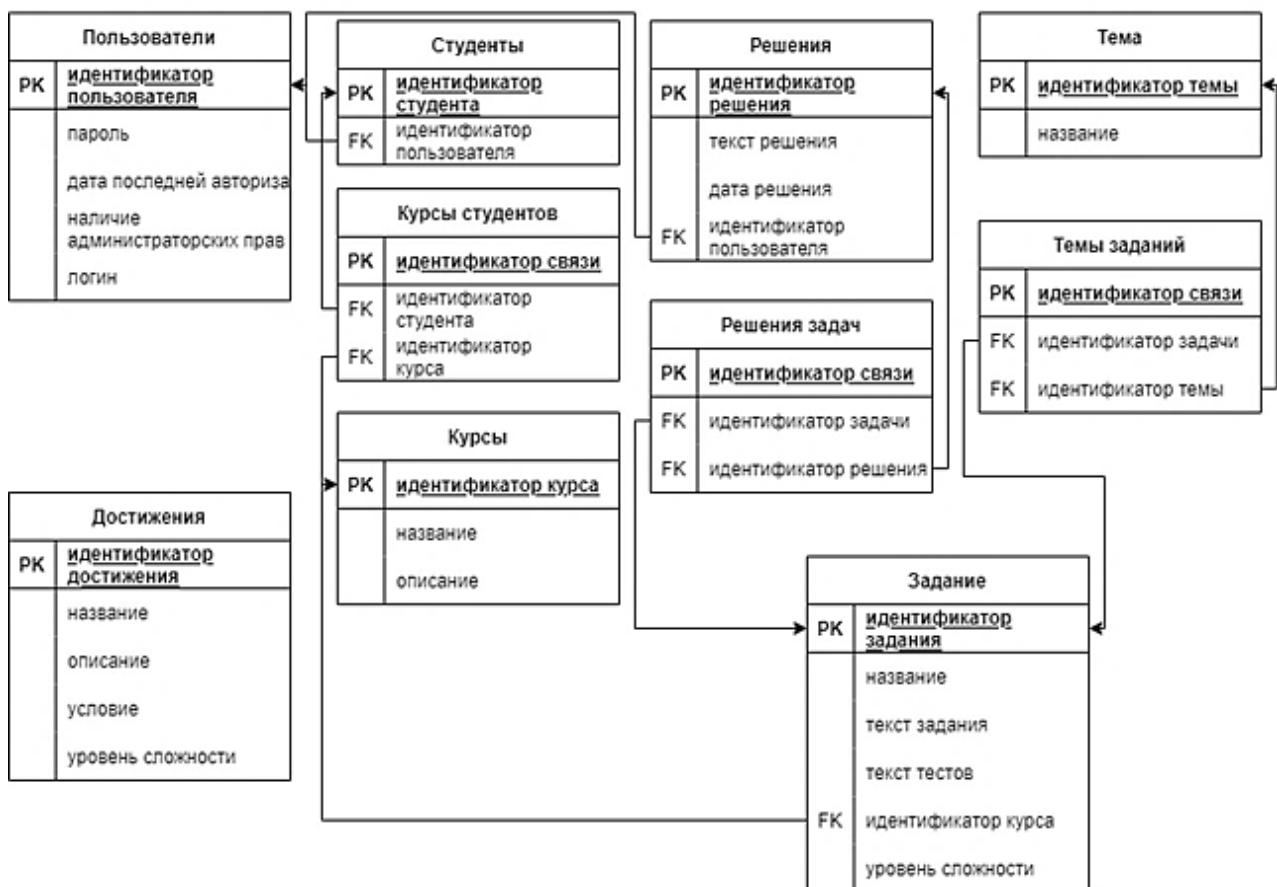


Рисунок. Общая схема логической структуры базы данных обучающего сервиса GameProg

- «Студенты» - содержит информацию о пользователях, относящихся к группе студентов. Каждое поле обладает следующими атрибутами: идентификатор студента (первичный ключ), идентификатор пользователя (вторичный ключ, ссылается на таблицу «Пользователи»);
- «Курсы» - содержит информацию о доступных учебных курсах. Каждое поле обладает следующими атрибутами: идентификатор курса (первичный ключ), название курса, краткое описание курса;
- «Курсы студентов» - содержит перечень курсов каждого студента. Каждое поле обладает следующими атрибутами: идентификатор связи (первичный ключ), идентификатор студента (вторичный ключ, ссылается на таблицу «Студенты»), идентификатор курса (вторичный ключ, ссылается на таблицу «Курсы»);
- «Тема» - содержит перечень концепций программирования, по которым на сервисе имеются задания. Каждое поле содержит следующие атрибуты: идентификатор темы (первичный ключ), краткое название темы;
- «Задание» - содержит основную информацию по каждой представленной на сервисе задаче. Каждое поле состоит из следующих атрибутов: идентификатор задания (первичный ключ), название задачи для отображения в списке задач, текст с описанием сути задачи, текст с перечнем тестов идентификатор курса (вторичный ключ, ссылается на таблицу «Курсы»), уровень сложности задачи;
- «Темы заданий» - содержит перечень тем по каждой имеющейся в базе задаче. Каждое поле состоит из следующих полей: идентификатор связи (первичный ключ), идентификатор задачи (вторичный ключ, ссылается на таблицу «Задание»), идентификатор темы (вторичный

ключ, ссылается к таблице «Тема»);

- «Решения» - содержит перечень хранящихся в базе решений студентов. Каждое поле состоит из следующих полей: идентификатор решения (первичный ключ), текст решения содержащий скрипт решения, дата предоставления решения, идентификатор пользователя (вторичный ключ, ссылается к таблице «Пользователи»);

- «Решения задач» - содержит перечень решений по каждой представленной на сервисе задаче. Каждое поле состоит из следующих полей: идентификатор связи (первичный ключ), идентификатор задачи (вторичный ключ, ссылается к таблице «Задание»), идентификатор решения (вторичный ключ, ссылается к таблице «Решения»);

- «Достижения» - содержит основную информацию по каждому представленному на сервисе достижению. Каждое поле состоит из следующих атрибутов: идентификатор достижения (первичный ключ), название достижения для отображения в списке достижения, текстовое описание достижения и условия его получения, условия для получения пользователем достижения, уровень сложности достижения.

В течение учебного года студенты факультета физико-математического образования, информатики и технологий проходили обучение языкам программирования в описанном выше сервисе. До и после использования сервиса у студентов измерялась степень заинтересованности в изучении программирования. Согласно результатам, средний уровень заинтересованности студентов увеличился на 25%; 80% участников оценили свой опыт работы с сервисом как положительный, особенно выделив такие элементы, как система достижений и соревновательная составляющая; 50% респондентов изначально слабо заинтересованных в изучении программирования, после опробования игрового обучающего сервиса GameProg начали проявлять большую заинтересованность; 40% изначально заинтересованных респондентов не изменили своего мнения; один респондент из десяти, имея изначально низкую заинтересованность, после опробования игрового сервиса не проявил повышения заинтересованности. Большинство участников отметили, что такие элементы, как система достижений и топ игроков, стимулируют к более активному изучению программирования.

При внедрении игровых элементов в образовательный процесс не обязательно ставить задачу полной геймификации процесса обучения с разработкой полноценной обучающей игры со всеми свойственными ей моментами: правилами, сюжетом, уровнями, наградами и т. д. Достаточно понимания принципов работы отдельных игровых элементов, для их успешного внедрения в процесс обучения, выделим *некоторые рекомендации*:

1. Используйте частичную геймификацию: внедрите игровые механики, такие как награды, достижения и рейтинги, чтобы стимулировать учащихся и улучшить их мотивацию.

2. Создайте задачи с возможностью экспериментирования: Предоставьте учащимся интерактивную среду, где они могут экспериментировать с кодом и видеть мгновенные результаты. Это помогает им лучше понять принципы программирования.

3. Включите соревновательный элемент: Организуйте соревнования или игровые сессии, где учащиеся могут соревноваться друг с другом или решать задачи в ограниченное время. Это помогает развить навыки работы под давлением и сроками.

4. Создавайте задания с постепенным усложнением: начинайте с простых задач и постепенно усложняйте их, чтобы учащиеся могли развивать свои навыки программирования постепенно.

5. Используйте визуальное представление: игровые элементы могут помочь

визуализировать абстрактные концепции программирования, делая их более доступными и понятными для учащихся.

6. Предоставляйте обратную связь: игровые элементы позволяют предоставлять мгновенную обратную связь об уровне выполнения заданий, что помогает учащимся лучше понимать свои успехи и области для улучшения.

7. Используйте коллаборативные игры: поощряйте учащихся работать в команде, используя многопользовательские игры или задания, чтобы они могли развивать навыки коллaborации и командной работы.

8. Поддерживайте интерес к игровой обучающей среде: обновляйте, дополняйте игровые элементы, чтобы сохранить интерес учащихся и предложить им новые вызовы, возможности.

Важно адаптировать эти рекомендации к целевой аудитории и целям обучения, чтобы создать эффективную и увлекательную обучающую среду для программирования.

Выходы

Исследование подчеркивает потенциал геймификации как преобразующего инструмента в образовании, особенно в таких областях, как программирование, где сложность обучения может привести к разобщенности. Структура GameProg, основанная на конкуренции и вознаграждении, поощряет настойчивость и постановку целей, что в конечном итоге способствует улучшению навыков обучения. В будущих исследованиях можно было бы дополнительно изучить, как различные типы игровых элементов влияют на результаты обучения и как такие сервисы можно масштабировать для более широкого использования в образовании. Предполагается, что внедрение сервиса GameProg в образовательные учреждения может не только повысить успеваемость, но и усилить интерес студентов к программированию. Ожидаемый эффект заключается в увеличении уровня практических навыков за счет регулярного выполнения игровых заданий и участия в соревновательных механизмах.

Проблема поддержания мотивации характерна не только для обучения программированию, но и для различных дисциплин, особенно в областях, требующих постоянной практики и концептуального понимания, таких как математика, инженерия или естественные науки. Успех геймифицированных подходов в обучении программированию, как продемонстрировало это исследование, может побудить преподавателей в других областях применять аналогичные методы для улучшения успеваемости учащихся.

Таким образом, можно заключить, что геймификация может стать мощным инструментом для повышения качества образовательного процесса в различных дисциплинах, а разработка и внедрение игровых обучающих сервисов, подобных GameProg, имеют большие перспективы для дальнейшего применения в образовательных учреждениях.

Литература

Ветушинский, А. С. (2020). Больше, чем просто средство: новый подход к пониманию геймификации. *Социология власти*, 3(32), 14–31.

Горлов, С. И. (2023). Методы обучения программированию ИТ бакалавров: игровые среды и программирование игр. *Педагогическая информатика*, (2), 191–195.

Гусев, И. Е. (2011). Исторические основы геймификации: игра, игровые технологии. *Проблемы современного педагогического образования*, (1(78)), 110–113.

Детердинг, С., Диксон, Д., Халед, Р. & Накке, Л. (2011). От элементов игрового дизайна к игровому процессу: определение геймификации. *Труды 15-й Международной академической конференции MindTrek по представлению будущих медиасред - MindTrek' (11)*, 9–15. [Электронный ресурс]. URL: <https://doi.org/10.1145/2181037.2181040>

Игнатьева, Э. А. & Никитин, В. В. (2022). Потенциал привнесения игровых элементов в образовательный процесс. *Цифровые технологии и инновации в развитии науки и образования*, 198–200.

Талызина, Н. Ф. (1969). Теоретические проблемы программированного обучения. МГУ (132).