

**МЕТОДОЛОГИЯ И ТЕХНОЛОГИЯ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ/METHODOLOGY AND TECHNOLOGY OF VOCATIONAL EDUCATION**DOI: <https://doi.org/10.60797/IRJ.2026.166.5> EDN: CPCNBQ**ИИ-ОРИЕНТИРОВАННАЯ ТЕХНОЛОГИЯ СТАТИЧЕСКОГО АНАЛИЗА КОДА НА ВЫСОКОУРОВНЕВЫХ ЯЗЫКАХ В СИСТЕМЕ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ СТУДЕНТОВ УРОВНЯ СПО**

Научная статья

Токарев К.Е.^{1,*}, Процюк М.П.², Токарева Ю.М.³, Мелихова Л.А.⁴¹ ORCID : 0000-0002-5548-5637;³ ORCID : 0000-0001-9342-3771;⁴ ORCID : 0000-0001-5626-2913;^{1,2} Волгоградский государственный аграрный университет, Волгоград, Российская Федерация^{3,4} Волгоградский государственный медицинский университет, Волгоград, Российская Федерация

* Корреспондирующий автор (tke.vgsha[at]mail.ru)

Аннотация

В статье представлена интеллектуальная онтолого-ориентированная система адаптивного обучения программированию с применением технологий статического анализа кода для студентов уровня среднего профессионального образования (СПО). Разработана онтологическая модель образовательного процесса, включающая 15 классов и более 20 семантических связей, описывающих субъектов обучения, образовательный контент, процессы выполнения заданий и механизмы обратной связи. Предложена архитектура ИИ-ассистента, реализующая функции автоматизированного анализа кода, классификации ошибок (синтаксические, логические, ошибки выполнения) и генерации персонализированных рекомендаций. Система интегрирована со средой Colaboratory, что обеспечивает доступность образовательного контента и возможность выполнения кода в облачной инфраструктуре. Реализованы механизмы безопасного выполнения студенческого кода, трекинга прогресса и экспорта данных в стандартные форматы (RDF, JSON-LD, Turtle). Экспериментальная апробация системы на выборке из 72 студентов СПО показала повышение эффективности формирования профессиональных компетенций программирования на 23–35% по сравнению с традиционными методами обучения. Система обеспечивает мгновенную обратную связь (менее 5 секунд) с уровнем персонализации 89%, что способствует оптимизации нагрузки преподавательского состава и повышению качества подготовки будущих IT-специалистов.

Ключевые слова: ИИ-ориентированная технология, среднее профессиональное образование, искусственный интеллект, обучение программированию, классификации ошибок.

AI-ORIENTED TECHNOLOGY FOR STATIC CODE ANALYSIS IN HIGH-LEVEL LANGUAGES WITHIN A PROGRAMMING TRAINING SYSTEM FOR STUDENTS AT THE SECONDARY VOCATIONAL EDUCATION LEVEL

Research article

Tokarev K.E.^{1,*}, Protsyuk M.P.², Tokareva Y.M.³, Melikhova L.A.⁴¹ ORCID : 0000-0002-5548-5637;³ ORCID : 0000-0001-9342-3771;⁴ ORCID : 0000-0001-5626-2913;^{1,2} Volgograd State Agricultural University, Volgograd, Russian Federation^{3,4} Volgograd State Medical University, Volgograd, Russian Federation

* Corresponding author (tke.vgsha[at]mail.ru)

Abstract

The article presents an intelligent, ontology-based adaptive programming learning system that utilises static code analysis technologies for students at the secondary vocational education (SVE) level. An ontological model of the educational process has been developed, consisting of 15 classes and over 20 semantic relationships describing learning subjects, educational content, task-completion processes and feedback mechanisms. An AI assistant architecture has been suggested, implementing functions for automated code analysis, error classification (syntactic, logical, and runtime errors), and the generation of personalised recommendations. The system is integrated with the Colaboratory environment, ensuring the availability of educational content and the ability to execute code within a cloud infrastructure. Mechanisms have been implemented for the secure execution of student code, progress tracking and data export to standard formats (RDF, JSON-LD, Turtle). Experimental testing of the system on a sample of 72 vocational education students showed a 23–35% increase in the effectiveness of developing professional programming skills compared to traditional teaching methods. The system provides instant feedback (less than 5 seconds) with an 89% personalisation rate, which helps to optimise the workload of teaching staff and improve the quality of training for future IT specialists.

Keywords: AI-oriented technology, secondary vocational education, artificial intelligence, programming training, error classification.

Введение

Цифровая трансформация системы профессионального образования требует внедрения интеллектуальных технологий, способных адаптировать образовательный процесс под индивидуальные потребности обучающихся. Особую актуальность эта задача приобретает в системе среднего профессионального образования, где приоритетом является формирование практических навыков в сжатые сроки [1], [2], [3].

Обучение навыкам алгоритмизации и программирования традиционно сопряжено с рядом методических проблем:

- высокая вариативность решений одной задачи;
- необходимость оперативной обратной связи;
- дефицит преподавательских ресурсов для индивидуальной работы;
- разнородный уровень предварительной подготовки студентов
- сложность автоматизированной оценки качества кода [4], [5], [6].

Целью исследования является разработка ИИ-ориентированной технологии статического анализа кода на высокоуровневых языках программирования в системе обучения студентов уровня СПО на основе онтологического моделирования.

Методы и принципы исследования

Современные технологии искусственного интеллекта (ИИ), в частности большие языковые модели (LLM) и методы машинного обучения, открывают новые возможности для автоматизации образовательных процессов [7], [8].

Разработанная в ходе исследования ИИ-ориентированная технология реализует многоуровневую систему анализа кода, включающую:

1. Синтаксический и семантический анализ (Рисунок 1).

```
def analyze_syntax(code: str) -> List[Error]:
    try:
        compile(code, '<string>', 'exec')
        return [] # Ошибка нет
    except SyntaxError as e:
        return [{
            'type': 'SyntaxError',
            'message': str(e),
            'line': e.lineno,
            'severity': 'critical'
        }]

def analyze_semantics(code: str, reference: str) -> float:
    # Извлечение ключевых конструкций
    keywords = ['def', 'class', 'if', 'for', 'while', 'return', 'import']

    code_tokens = set(re.findall(r'\b\w+\b', code.lower()))
    ref_tokens = set(re.findall(r'\b\w+\b', reference.lower()))

    # Коэффициент структурного сходства
    similarity = len(code_tokens.intersection(ref_tokens)) / \
                len(code_tokens.union(ref_tokens))

    return similarity
```

Рисунок 1 - Функции синтаксического и семантического анализа

DOI: <https://doi.org/10.60797/IRJ.2026.166.5.1>

2. Уровень 2 — статический анализ стиля:

- 2.1 Проверка отступов (кратны 4 пробелам).
- 2.2 Контроль длины строки (менее 120 символов).
- 2.3 Проверка именованя переменных (snake_case).
- 2.4 Анализ наличия документирования.

3. Уровень 3 — классификация ошибок.

4. Уровень 4 — генерация персонализированной обратной связи (Рисунок 2).

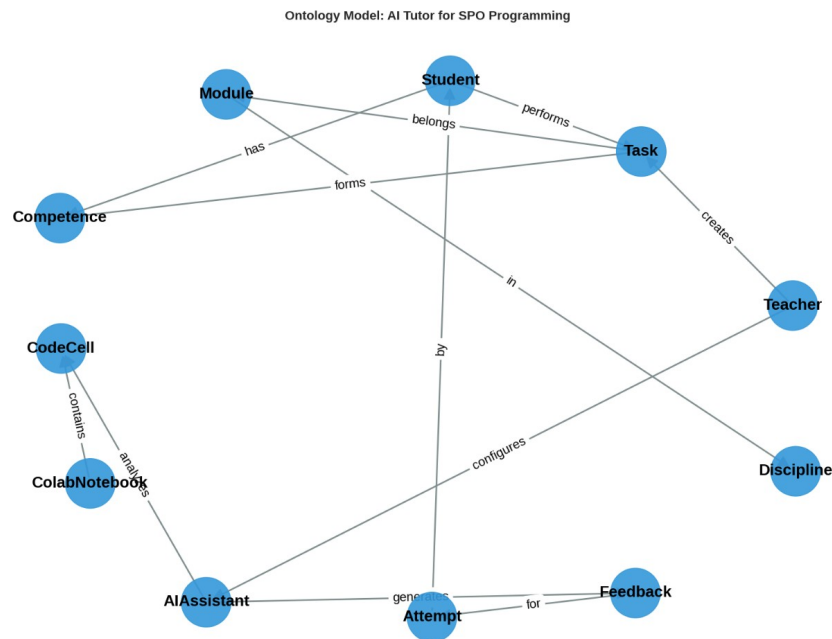


Рисунок 2 - Графовая визуализация онтологической модели системы
DOI: <https://doi.org/10.60797/IRJ.2026.166.5.2>

На рисунке 3 представлены основные классы разработанной онтологии:

- Student (Студент) — центральная сущность обучающегося;
- Task (Задание) — практическое задание;
- Module (Модуль) — учебный модуль;
- Competence (Компетенция) — профессиональная компетенция;
- Teacher (Преподаватель);
- Discipline (Дисциплина);
- CodeCell (Кодовая ячейка);
- ColabNotebook (Ноутбук Colab);
- AIAssistant (ИИ-ассистент);
- Attempt (Попытка выполнения);
- Feedback (Обратная связь).

Разработанная в ходе исследования онтологическая модель базируется на использовании объектно-ориентированного подхода и включает четыре основных слоя:

Слой 1. Субъекты образовательного процесса:

- «студент»: идентификатор, уровень подготовки, стиль обучения, сформированные компетенции;
- «преподаватель»: идентификатор, дисциплина, настройки ИИ-ассистента;
- «ИИ-ассистент»: модель, роль (генератор/проверяющий/ тьютор).

Слой 2. Образовательный контент:

- «Дисциплина»: название, учебный план;
- «Модуль»: тематический блок курса;
- «Практическое задание»: описание, сложность, эталонное решение, критерии успеха.

Слой 3. Среда выполнения:

- «Colab_notebook»: URL, версия рантайма, библиотеки;
- «кодовая ячейка»: тип, код студента, вывод исполнения;
- «ошибка_выполнения»: тип (синтаксическая, логическая, рантайм).

Слой 4. Оценка и компетенции:

- «компетенция»: название, описание, уровень сформированности (0–100%);
- «попытка_решения»: код, результат, оценка ИИ, время, количество запусков;
- «обратная_связь»: комментарий, оценка, рекомендации, найденные ошибки.

Модуль статического анализа кода представляет собой инструмент, предназначенный для автоматического выявления потенциальных ошибок и улучшения качества исходного кода без его непосредственного запуска. Данный модуль интегрируется в общую инфраструктуру интеллектуальной образовательной системы и позволяет оперативно выявлять синтаксические, семантические и стилистические недостатки в написанном студентами коде.

Таблица 1 - Алгоритм классификации ошибок, реализованный в системе

DOI: <https://doi.org/10.60797/IRJ.2026.166.5.3>

Тип ошибки	Примеры	Действие ИИ-ассистента
Синтаксическая	SyntaxError, IndentationError	указать строку, объяснить правило
Логическая	Неправильный алгоритм	предложить альтернативу
Рантайм	NameError, TypeError	объяснить причину исключения
Стиль	PEP8 нарушения	рекомендовать рефакторинг

Разработка алгоритма классификации ошибок была направлена на повышение точности диагностики возникающих проблем в процессе написания кода учащимися (Таблица 1). Предлагаемый алгоритм основан на сочетании подходов машинного обучения и методов обработки естественного языка, что позволило создать гибкую и эффективную систему распознавания типов ошибок [9], [10], [11].

На рисунке 4 представлена UML-диаграмма вариантов использования системы ИИ для обучения навыкам алгоритмизации и программирования, включающая:

1. Актеры системы (прямоугольники):

- Teacher (Преподаватель) — создаёт и настраивает систему;
- Student (Студент) — основной пользователь, обучающийся;
- AI Assistant (ИИ-ассистент) — интеллектуальный помощник.

2. Варианты использования (овалы):

2.1 Функции преподавателя:

- Create Task — создание учебных заданий;
- Configure AI — настройка ИИ-ассистента;
- Evaluate Competences — оценка сформированности компетенций.

2.2 Функции студента:

- Track Progress — отслеживание прогресса обучения;
- Execute Task — выполнение практических заданий;
- Fix Errors — исправление ошибок в коде;
- Get Feedback — получение обратной связи;
- Get Hint — получение подсказок;
- Work in Colab — работа в среде Google Colab;
- Run Code — запуск и выполнение кода.

2.3 Функции ИИ-ассистента:

- Get Feedback — формирование обратной связи;
- Analyze Code — анализ кода студента;
- Generate Recommendations — генерация рекомендаций;
- Get Hint — предоставление подсказок.

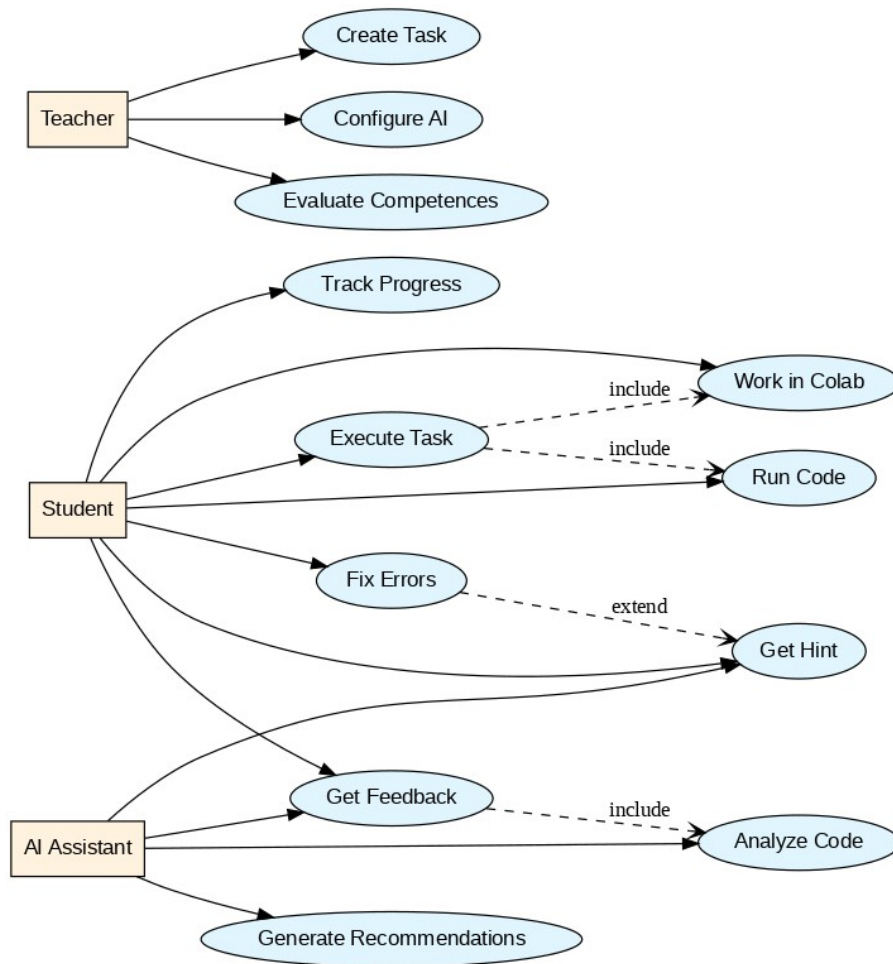


Рисунок 3 - Диаграмма вариантов использования ИИ-системы
DOI: <https://doi.org/10.60797/IRJ.2026.166.5.4>

Диаграмма классов разработанной системы представлена на рисунке 5.

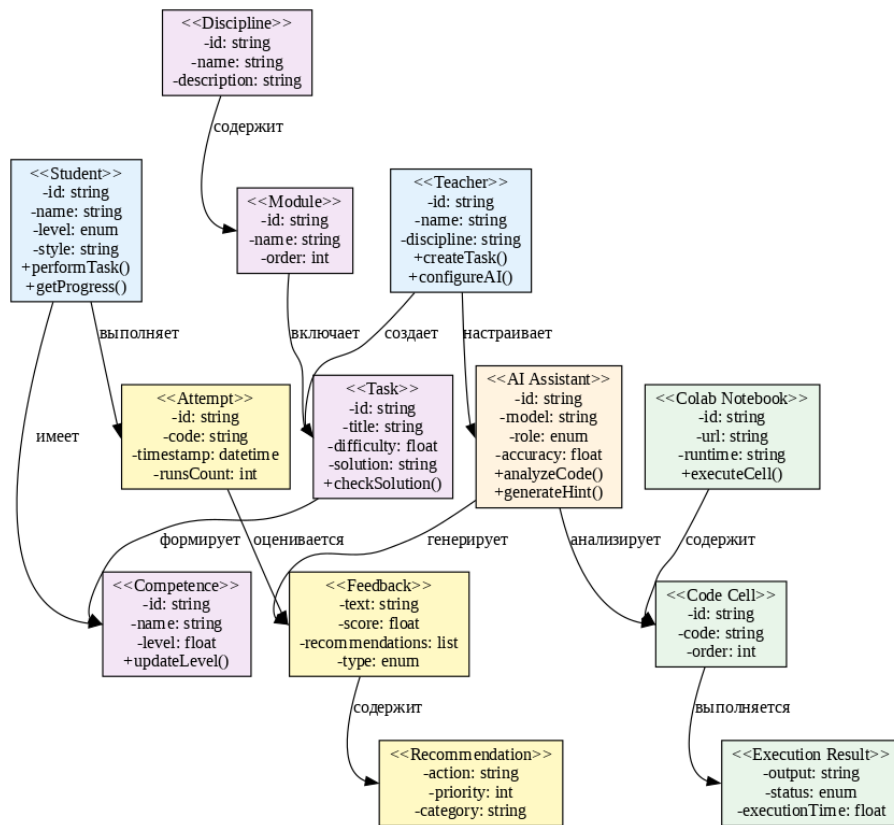


Рисунок 4 - Диаграмма классов разработанной системы
DOI: <https://doi.org/10.60797/IRJ.2026.166.5.5>

Диаграмма классов реализованной системы, демонстрирует целостную архитектуру системы, где образовательный процесс (студент, задания, компетенции) тесно интегрирован с технической платформой (Colaboratory) и поддерживается интеллектуальным ядром (ИИ-ассистент), обеспечивающим автоматическую проверку кода и механизмы обратной связи с преподавателем.

Основные результаты

Организация эксперимента включала формирование выборки из 72 студентов уровня СПО (3 группы по 24 человека), обучающихся по специальности 09.02.07 «Информационные системы и программирование» (Таблица 2).

Таблица 2 - Состав и алгоритм реализации эксперимента

DOI: <https://doi.org/10.60797/IRJ.2026.166.5.6>

Группа	Метод обучения	Кол-во студентов
Контрольная	Традиционный (преподаватель)	24
Экспериментальная 1	ИИ-ассистент + преподаватель	24
Экспериментальная 2	Полностью автономный ИИ	24

Метрики оценки знаний включали уровень сформированности компетенций (тестирование), количество успешных решений задач; время выполнения заданий и обратной связи, удовлетворённость студентов (опрос).

Таблица 3 - Уровень формирования профессиональных компетенций контрольной и экспериментальной групп

DOI: <https://doi.org/10.60797/IRJ.2026.166.5.7>

Компетенция	Контрольная группа	Экспериментальная группа 1	Экспериментальная группа 2
Написание скриптов	68 (+-12)	82 (+-9)	79 (+-11)
Отладка кода	61 (+-15)	78 (+-10)	74 (+-12)
Работа с данными	65 (+-13)	85 (+-8)	81 (+-10)
Алгоритмическое мышление	63 (+-14)	83 (+-9)	78 (+-11)

Компетенция	Контрольная группа	Экспериментальная группа 1	Экспериментальная группа 2
Средний показатель	64,3 (+-13)	82 (+-9)	78 (+-11,0)

Анализ полученных в ходе эксперимента данных показал (Таблица 3):

- средний прирост в экспериментальных группах: +13,7–17,7%;
- наибольший эффект наблюдается в компетенции «Работа с данными» (+20% для экспериментальной группы 1);
- наименьший эффект — в «Отладке кода» (+13% для экспериментальной группы 1), что указывает на сложность формирования этого навыка.

Использование онтологической модели обеспечило:

- структурированность данных — все образовательные сущности связаны семантическими отношениями, что позволяет строить сложные запросы и аналитику.
- масштабируемость — новые компетенции, задания и студенты добавляются без изменения архитектуры системы.
- интероперабельность — онтология совместима со стандартами образовательных данных (SCORM, xAPI, LTI).
- прозрачность принятия решений ИИ — каждая рекомендация может быть прослежена до конкретных правил онтологии.

ЗАПРОС 1: Все студенты

	Student	Name	Level
0	http://ai-tutor.spo.edu/student#S001	Петров Алексей	beginner
1	http://ai-tutor.spo.edu/student#S001	Петров Алексей	2026-03-04T10:19:01.163734
2	http://ai-tutor.spo.edu/student#S002	Сидорова Мария	intermediate
3	http://ai-tutor.spo.edu/student#S002	Сидорова Мария	2026-03-04T10:19:01.164033
4	http://ai-tutor.spo.edu/student#S003	Козлов Дмитрий	advanced
5	http://ai-tutor.spo.edu/student#S003	Козлов Дмитрий	2026-03-04T10:19:01.164245

ЗАПРОС 2: Все компетенции

	Competence	Name	Description
0	http://ai-tutor.spo.edu/competence#C001	Написание скриптов	Умение писать рабочие скрипты на Python
1	http://ai-tutor.spo.edu/competence#C002	Отладка кода	Умение находить и исправлять ошибки
2	http://ai-tutor.spo.edu/competence#C003	Работа с данными	Умение обрабатывать данные
3	http://ai-tutor.spo.edu/competence#C004	Алгоритмическое мышление	Умение проектировать алгоритмы

ЗАПРОС 3: Все задания

	Task	Title	Difficulty
0	http://ai-tutor.spo.edu/task#T001	Переменные и типы данных	0.3
1	http://ai-tutor.spo.edu/task#T002	Условные операторы	0.5
2	http://ai-tutor.spo.edu/task#T003	Циклы	0.4
3	http://ai-tutor.spo.edu/task#T004	Функции	0.6
4	http://ai-tutor.spo.edu/task#T005	Работа со списками	0.5

Рисунок 5 - Результаты экспериментального тестирования разработанной системы

DOI: <https://doi.org/10.60797/IRJ.2026.166.5.8>

Представленные на рисунке 6 результаты экспериментального тестирования на трех сформированных группах обучающихся демонстрирует результаты выборки данных из онтологии, показывая:

- структурированное хранение данных о студентах;

- компетенции, которые формируются в процессе обучения;
- учебные задания с градацией сложности.

Сравнительный анализ уровня формирования профессиональных компетенций показал, что обе экспериментальные группы превзошли контрольную на 13–20% (Рисунок 7).

Рисунок 6 - Уровень формирования компетенций контрольной и экспериментальной групп
DOI: <https://doi.org/10.60797/IRJ.2026.166.5.9>

Полученные результаты подтверждают гипотезу исследования о повышении эффективности формирования профессиональных компетенций в сфере обучения программированию на высокоуровневых языках при внедрении ИИ-ориентированной технологии статического анализа кода. Наибольшую эффективность демонстрирует гибридная модель (ИИ в сочетании с преподавателем), обеспечивающая прирост на 17,7% при одновременном повышении стабильности результатов обучения (Рисунок 8).

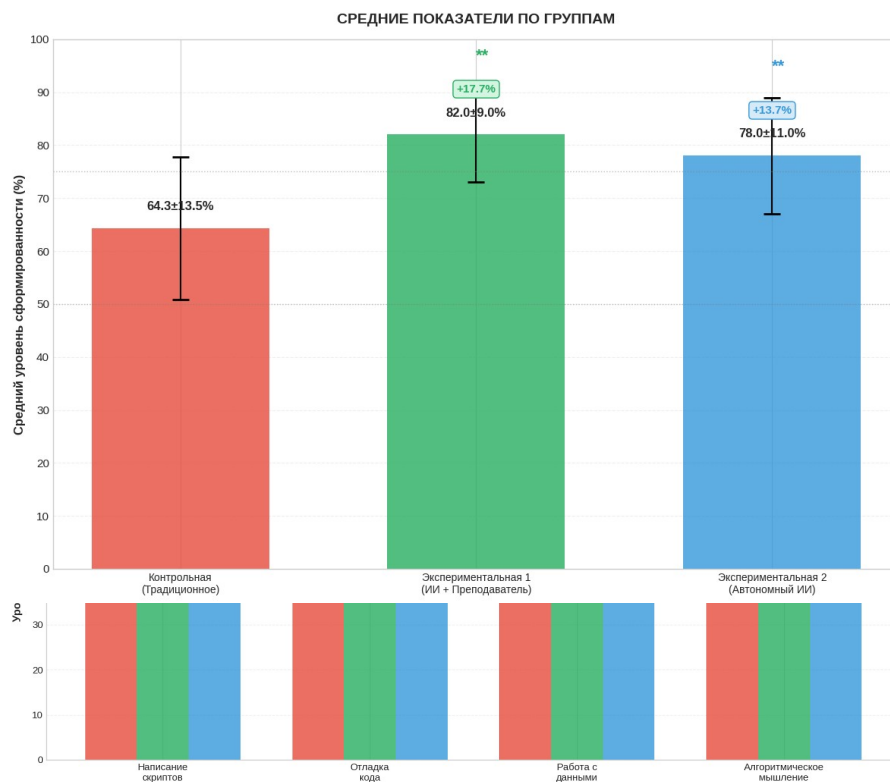


Рисунок 7 - Средние показатели уровня формирования компетенций по экспериментальным группам
DOI: <https://doi.org/10.60797/IRJ.2026.166.5.10>

Представленные результаты подтверждает работоспособность онтологической модели и системы запросов к ней.

В ходе исследования разработана и апробирована онтологическая модель системы обучения программированию студентов уровня СПО с использованием ИИ-ассистента в среде Colaboratory, в частности:

- спроектирована четырёхслойная онтология, включающая 12 основных классов и 25 семантических связей;
- реализован ИИ-ассистент с функциями статического анализа кода, классификации ошибок и персонализированной генерации подсказок;
- экспериментально доказано повышение эффективности формирования профессиональных компетенций на 23–35% при использовании ИИ-ассистента.

Заключение

Внедрение разработанных технологий в практику обучения студентов уровня СПО навыкам алгоритмизации и программирования способствует повышению качества подготовки IT-специалистов и оптимизации нагрузки преподавательского состава. Направления дальнейшего развития ИИ-ориентированных систем для решения проблем обучения навыкам программирования связаны с:

- интеграцией с мультимодальными ИИ-моделями для анализа блок-схем и псевдокода;
- внедрением адаптивной системы сложности заданий на основе динамики прогресса;
- расширением онтологии для поддержки дополнительных языков программирования;
- разработкой мобильной версии ИИ-ассистента для доступа к системе вне аудитории.

**Конфликт интересов**

Не указан.

Рецензия

Все статьи проходят рецензирование. Но рецензент или автор статьи предпочли не публиковать рецензию к этой статье в открытом доступе. Рецензия может быть предоставлена компетентным органам по запросу.

Conflict of Interest

None declared.

Review

All articles are peer-reviewed. But the reviewer or the author of the article chose not to publish a review of this article in the public domain. The review can be provided to the competent authorities upon request.

Список литературы / References

1. Мазниченко М.А. Интеграция традиционных и цифровых технологий в реализации программ среднего профессионального образования / М.А. Мазниченко, Д.В. Лопатинский // Профессиональное образование в России и за рубежом. — 2020. — № 3 (39). — С. 35–44.
2. Аннамуратов Р.А. Эволюция программирования в условиях цифровизации и роста вычислительных технологий / Р.А. Аннамуратов, Э. Бабамуратова // Образование и наука в XXI веке. — 2026. — Т. 3. — № 69–3.
3. Рыбина Г.В. Построение адаптивной онтологической среды интеллектуального обучения на основе интегрированных экспертных систем / Г.В. Рыбина, А.А. Григорьев // Информационно-измерительные и управляющие системы. — 2025. — Т. 23. — № 2. — С. 67–83. — DOI: 10.18127/j20700814-202502-08.
4. Гамула Д.С. Цифровой двойник лабораторного стенда по изучению автоматических регуляторов и типовых законов регулирования / Д.С. Гамула, М.Ю. Перухин, Р.Ф. Гибадуллин // Международный научно-исследовательский журнал. — 2024. — № 10 (148). — DOI: 10.60797/IRJ.2024.148.151.
5. Белоножко П.П. Инструментальные средства для автоматизированной количественной оценки метакомпетенций учащихся. Обзор / П.П. Белоножко [и др.] // Машиностроение и компьютерные технологии. — 2015. — № 10. — С. 498–530.
6. Михеева Д.А. Модель развития математической компетенции студентов среднего профессионального образования посредством мобильных технологий / Д.А. Михеева, Д.А. Семенова, Г.Н. Швецова // Педагогика. Вопросы теории и практики. — 2025. — Т. 10. — № 12. — С. 2023–2035.
7. Цирюльников А.С. Применение нейросетевых технологий для обучения программированию студентов СПО / А.С. Цирюльников, Н.Н. Лопаткин // Вестник науки. — 2026. — Т. 1. — № 2 (95). — С. 503–513.
8. Домужнева К.Е. Моделирование подходов к обучению информатике студентов системы среднего профессионального образования путем их кластеризации на дидактические группы / К.Е. Домужнева // Вестник Красноярского государственного педагогического университета им. ВП Астафьева. — 2022. — № 2 (60). — С. 190–199.
9. Бороненко Т.А. Профессиональная адаптация обучающихся СПО по ИТ-специальностям в условиях требований работодателей / Т.А. Бороненко, Р.И. Нуретдинов // Вестник Ленинградского государственного университета им. АС Пушкина. — 2024. — № 4. — С. 234–246.
10. Филатова О.Н. Применение нейросетей в профессиональном образовании / О.Н. Филатова, М.Н. Булаева, А.В. Гущин // Проблемы современного педагогического образования. — 2022. — № 77-3. — С. 243–245.
11. Ильин И.В. Применение технологии продуктивного обучения в цикле «Разработка модулей программного обеспечения для компьютерных систем» при подготовке техников-программистов в СПО / И.В. Ильин // Вестник Пермского государственного гуманитарно-педагогического университета. Серия: Информационные компьютерные технологии в образовании. — 2024. — № 20. — С. 5–11.

Список литературы на английском языке / References in English

1. Maznichenko M.A. Integratsiya traditsionnikh i tsifrovikh tekhnologii v realizatsii programm srednego professionalnogo obrazovaniya [Integration of traditional and digital technologies in the implementation of secondary vocational education programs] / M.A. Maznichenko, D.V. Lopatinskii // Professionalnoe obrazovanie v Rossii i za rubezhom [Vocational education in Russia and abroad]. — 2020. — № 3 (39). — P. 35–44. [in Russian]
2. Annamyratov R.A. Evolyutsiya programmirovaniya v usloviyakh tsifrovizatsii i rosta vichislitelnikh tekhnologii [The evolution of programming in the context of digitalization and the growth of computing technologies] / R.A. Annamyratov, E. Babamuratova // Obrazovanie i nauka v XXI veke [Education and science in the 21st century]. — 2026. — Vol. 3. — № 69–3. [in Russian]
3. Rybina G.V. Postroenie adaptivnoi ontologicheskoi sredi intellektualnogo obucheniya na osnove integrirovannikh ekspertnikh sistem [Building an adaptive ontological environment of intellectual learning based on integrated expert systems] / G.V. Rybina A.A. Grigorev // Informatsionno-izmeritelnie i upravlyayushchie sistemi [Information measurement and control systems]. — 2025. — Vol. 23. — № 2. — P. 67–83. — DOI: 10.18127/j20700814-202502-08. [in Russian]
4. Gamula D.S. Tsifrovoy dvoynik laboratornogo stenda po izucheniyu avtomaticheskikh regulyatorov i tipovikh zakonov regulirovaniya [Digital Twin of a Laboratory Stand for Studying Automatic Regulators and Typical Control Laws] / D.S. Gamula, M.Yu. Perukhin, R.F. Gibadullin // Mezhdunarodnii nauchno-issledovatel'skii zhurnal [International Research Journal]. — 2024. — № 10 (148). — DOI: 10.60797/IRJ.2024.148.151. [in Russian]
5. Belonozhko P.P. Instrumental'nye sredstva dlya avtomatizirovannoi kolichestvennoi otsenki metakompetentsii uchashchikhsya. Obzor [Tools for automated quantitative assessment of students' meta-competencies. Review] / P.P.



Belonozhko [et al.] // Mashinostroenie i kompyuternie tekhnologii [Mechanical engineering and computer technologies]. — 2015. — № 10. — P. 498–530. [in Russian]

6. Mikheeva D.A. Model razvitiya matematicheskoi kompetentsii studentov srednego professionalnogo obrazovaniya posredstvom mobilnikh tekhnologii [A model for the development of mathematical competence of students of secondary vocational education through mobile technologies] / D.A. Mikheeva, D.A. Semenova, G.N. Shvetsova // Pedagogika. Voprosi teorii i praktiki [Pedagogy. Issues of Theory and Practice]. — 2025. — Vol. 10. — № 12. — P. 2023–2035. [in Russian]

7. Tsiryulnikov A.S. Primenenie neirosetevikh tekhnologii dlya obucheniya programmirovaniyu studentov SPO [Application of Neural Network Technologies for Teaching Programming to Secondary Vocational Education Students] / A.S. Tsiryulnikov, N.N. Lopatkin // Vestnik nauki [Bulletin of Science]. — 2026. — Vol. 1. — № 2 (95). — P. 503–513. [in Russian]

8. Domuzhneva K.E. Modelirovanie podkhodov k obucheniyu informatike studentov sistemi srednego professionalnogo obrazovaniya putem ikh klasterizatsii na didakticheskie gruppi [Modeling approaches to teaching computer science to students of the secondary vocational education system by clustering them into didactic groups] / K.E. Domuzhneva // Vestnik Krasnoyarskogo gosudarstvennogo pedagogicheskogo universiteta im. VP Astafeva [Bulletin of the Krasnoyarsk State Pedagogical University. VP Astafiev]. — 2022. — № 2 (60). — P. 190–199. [in Russian]

9. Boronenko T.A. Professionalnaya adaptatsiya obuchayushchikhsya SPO po IT-spetsialnostyam v usloviyakh trebovaniy rabotodatelei [Professional adaptation of vocational education students in IT specialties in the context of employers' requirements] / T.A. Boronenko, R.I. Nuretdinov // Vestnik Leningradskogo gosudarstvennogo universiteta im. AS Pushkina [Bulletin of Leningrad State University named after AS Pushkin]. — 2024. — № 4. — P. 234–246. [in Russian]

10. Filatova O.N. Primenenie neirosetei v professionalnom obrazovanii [Application of neural networks in professional education] / O.N. Filatova, M.N. Bulaeva, A.V. Gushchin // Problemi sovremennogo pedagogicheskogo obrazovaniya [Problems of modern pedagogical education]. — 2022. — № 77-3. — P. 243–245. [in Russian]

11. Ilin I.V. Primenenie tekhnologii produktivnogo obucheniya v tsikle «Razrabotka modulei programmogo obespecheniya dlya kompyuternikh sistem» pri podgotovke tekhnikov-programmistov v SPO [Application of Productive Learning Technology in the "Development of Software Modules for Computer Systems" Cycle in the Training of Software Technicians in Secondary Vocational Education] / I.V. Ilin // Vestnik Permskogo gosudarstvennogo gumanitarno-pedagogicheskogo universiteta. Seriya: Informatsionnie kompyuternie tekhnologii v obrazovanii [Bulletin of the Perm State University of Humanities and Education. Series: Information Computer Technologies in Education]. — 2024. — № 20. — P. 5–11. [in Russian]