УДК 004.89, 005.8

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ПРОЕКТАМИ С ИНТЕЛЛЕКТУАЛЬНЫМ АССИСТЕНТОМ

Карпейкин Григорий Сергеевич[1], Иванцова Ольга Владимировна[2]

[1]Студент;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: kgs.22@uni-dubna.ru.

[2]Старший преподаватель;
Государственный университет «Дубна»;
Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;
e-mail: o_olga@uni-dubna.ru.

В работе исследуется проблема повышения эффективности проектного управления в малых командах сферы информационных технологий и при индивидуальной разработке программного обеспечения в условиях отсутствия выделенных менеджерских ролей. На основе сравнительного анализа существующих решений обоснована необходимость создания специализированного инструментария поддержки принятия решений. Разработана архитектура и программная реализация информационной системы управления проектами, ключевым элементом которой выступает интеллектуальный ассистент, спроектированный с использованием агентного подхода и реализующий функции диалогового взаимодействия с пользователем, декомпозиции целей проектов и динамической генерации описания проекта. В работе определена роль интеллектуального ассистента в проектном управлении, сформированы перспективы развития предложенного решения, включающие переход к мультиагентному взаимодействию, локальное развертывание больших языковых моделей и интеграцию с внешними сервисами.

Ключевые слова: управление проектами, искусственный интеллект, ИИ-ассистент, агентная архитектура, микросервисная архитектура.

Для цитирования:

Карпейкин Г. С., Иванцова О. В. Разработка системы управления проектами с интеллектуальным ассистентом // Системный анализ в науке и образовании. 2026. № 3. С. 13–21. EDN: DEBWEG. URL: https://sanse.ru/index.php/sanse/article/view/732.

DEVELOPMENT OF A PROJECT MANAGEMENT SYSTEM WITH AN INTELLIGENT ASSISTANT

Karpeykin Grigoriy S.[1], Ivantsova Olga V.[2]

[1]Student;
Dubna State University;
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;
e-mail: kgs.22@uni-dubna.ru.

[2]Senior teacher;
Dubna State University;
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;
e-mail: o_olga@uni-dubna.ru

This paper examines the problem of improving project management efficiency in small IT teams and individual software development in the absence of dedicated management roles. A comparative analysis of existing solutions substantiates the need to create specialized decision support tools. The architecture and software implementation of a project management information system have been developed, the key element

Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) https://creativecommons.org/licenses/by/4.0/deed.ru

of which is an intelligent assistant designed using an agent-based approach and implementing functions for dialogic interaction with the user, decomposition of project goals, and dynamic generation of project descriptions. The paper defines the role of an intelligent assistant in project management and outlines development prospects for the proposed solution, including the transition to multi-agent interaction, local deployment of large language models, and integration with external services.

Keywords: project management, artificial intelligence, AI assistant, agent architecture, microservice architecture.

For citation:

Karpeykin G. S., Ivantsova O. V. Development of a project management system with an intelligent assistant. *System analysis in science and education*, 2026;(3):13-21 (in Russ). EDN: DEBWEG. Available from: https://sanse.ru/index.php/sanse/article/view/732.

Введение

Современный этап развития информационных технологий характеризуется ростом сложности проектов и усилением конкурентной динамики. В этих условиях эффективное управление проектами становится критическим фактором успеха, определяющим не только своевременность и качество результата, но и саму жизнеспособность инициативы. Применение гибких методологий (*Agile*, *Scrum*, *Kanban*), обеспечивая адаптивность, одновременно предъявляют повышенные требования к оперативному планированию, декомпозиции целей и непрерывному мониторингу параметров проекта [1].

Наиболее значимыми данные вызовы становятся для малых проектных команд и индивидуальных разработчиков, чьи ресурсы ограничены, а необходимость совмещения множества профессиональных ролей приводит к значительным непроизводственным затратам времени. Если в крупных организациях задачи декомпозиции целей и системной аналитики распределяются между выделенными менеджерскими ролями (проектными менеджерами и бизнес-аналитиками), то в малых командах данные роли, как правило, экономически нецелесообразны [2],[8]. Поэтому индивидуальному разработчику приходится совмещать функции аналитика, архитектора и тестировщика, что повышает требования к инструментам организации работы. Дополнительной проблемой является необходимость проектирования плана и разбиения проекта на задачи, требующая значительных временных затрат и определенных знаний. Это может привести к отказу от развития проекта на стадии проектирования: проект остается нереализованным в виде идеи, так и не перейдя в стадию разработки [3].

Существующие программные комплексы автоматизации проектной деятельности (в частности, *Jira*, *Trello*, *YouGile*, *ClickUp*) представляют собой развитые инструменты, ориентированы преимущественно на организационную поддержку процессов (ведение списков задач, мониторинг испол-нителей) и не предлагают интеллектуального сопровождения процедур планирова-ния и системного анализа. Таким образом, формируется объективная научно-практическая потребность в создании автоматизированных систем управления, сочетающих функции классического планировщика задач и интеллектуального ас-систента поддержки принятия решений, способного частично автоматизировать функции аналитика и менеджера [4].

1. Анализ существующих решений

На основе проведенного анализа предметной области выявлена объективная потребность в инструментарии, обеспечивающем поддержку малых проектных групп и индивидуальных разработчиков на всех этапах жизненного цикла программного обеспечения. С целью верификации функциональных возможностей и ограничений современных программных комплексов управления проектами далее приведен критический анализ наиболее распространенных решений (*Jira*, *Trello*, *YouGile*, *ClickUp*) с акцентом на специфику их применения в условиях малых команд и степень интеграции интеллектуальных подсистем [4].

Программный комплекс *Jira* ориентированный на средние и крупные команды разработчиков, работающие по гибким методологиям, предлагает настраиваемые рабочие процессы, детальную отчетность и глубокую интеграцию с продуктами *Atlassian*. Однако гибкость настройки оборачивается

высоким порогом входа: первоначальное освоение требует значительного времени, для полноценного администрирования нередко нужен выделенный специалист. Интеллектуальная поддержка планирования в *Jira* представлена продуктом *Rovo*, однако доступна только в платных тарифах – декомпозиция целей и выявление рисков в бесплатной версии возлагаются на пользователя. Платформа предоставляет инструментарий, но не помогает в аналитической работе без приобретения платной подписки.

Платформа *Trello* базируется на методологии *Kanban* и представляет собой инструмент визуализации потока задач. Минималистичный интерфейс, основанный на концепции карточек и досок, обеспечивает низкие барьеры освоения в условиях малых команд. Внедренная подсистема автоматизации (*Trello Assistant*) ориентирована на выполнение рутинных операций (генерация шаблонных описаний, суммаризация текстового контента), однако ее функциональные возможности ограничены. Данная подсистема реализует алгоритмы жесткой автоматизации, но не обладает свойствами интеллектуального ассистента, так как не способна осуществлять семантическую декомпозицию целей или многокритериальную оценку рисков. Концептуальные ограничения архитектуры *Trello* также выражаются в отсутствии штатного инструментария итерационного планирования и явного моделирования взаимосвязей и зависимостей между задачами.

Платформа *YouGile* представляет собой российское решение для управления проектами, ориентированное на локальный рынок и сочетающее визуальную наглядность с базовым инструментарием для организации командной работы. Доступные пользователю функциональные элементы включают канбан-доску, диаграмму Ганта, встроенные чаты для обсуждения задач и гибкую систему прав доступа. Однако экосистема интеграций существенно уступает зарубежным аналогам, а встроенный ИИ-ассистент доступен в *beta*-версии, однако его возможности носят общий характер – специализированная декомпозиция целей и интеллектуальный анализ рисков не реализованы. Платформа предоставляет базовый управленческий инструментарий, но не сокращает аналитическую нагрузку на технического специалиста.

Платформа *ClickUp* позиционируется как универсальное решение, объединяющее средства управления проектами, документирования, мониторинга времени и аналитики загрузки команды. Набор доступных инструментов включает канбан-доски, диаграммы Ганта, календари и сценарии автоматизации рутинных операций. Однако большинство функциональных возможностей жестко привязана к тарифным планам: интеллектуальные функции и расширенная автоматизация требуют платной подписки. Данное обстоятельство накладывает дополнительные финансовые ограничения на малые команды с ограниченным бюджетом.

Результаты сравнительного анализа систематизированы (см. табл. 1), где представлены ключевые характеристики платформ по следующим аспектам: поддержка малых команд, базовый функционал управления проектами и наличие интеллектуального функционала.

Табл. 1. Сравнительный анализ систем управления проектами

Функции	***Jira***	***Trello***	***YouGile***	***ClickUp***
Малые команды (1–9 чел.)	Частично	Да	Да	Да
Создание и структурирование проектов	Да	Да	Да	Да
Гибкая система ролей и прав	Да	Частично	Да	Платно
ИИ-ассистент	Платно	Частично	Да	Платно
Декомпозиция целей (ИИ)	Нет	Нет	Нет	Платно
Генерация описаний (ИИ)	Нет	Платно	Нет	Платно
Открытый исходный код	Нет	Нет	Нет	Нет
Бесплатный ИИ-функционал	Нет	Нет	Нет	Нет

Как видно из таблицы, ни одна из рассмотренных систем не предоставляет инетллектуальной поддержки в бесплатной версии. Декомпозиция целей либо отсутствует, либо доступна только в

платных тарифах. Таким образом, на рынке обнаруживается потребность в сбалансированном решении, которое сочетает достаточную для малых команд функциональность управления проектами с бесплатным доступом к интеллектуальным функциям [4].

2. Архитектура и реализация системы

Разработанная система «НейроТаск» представляет собой веб-приложение для управления проектами с интегрированным интеллектуальным ассистентом, ориентированное на небольшие команды и индивидуальных разработчиков. Система обеспечивает полный цикл управления проектами – от создания и настройки ролевой модели до контроля исполнения задач с визуализацией на канбан-доске и диаграмме Ганта. После авторизации пользователь попадает на главную страницу с приветственным экраном и навигацией к основным разделам системы (см. рис. 1).

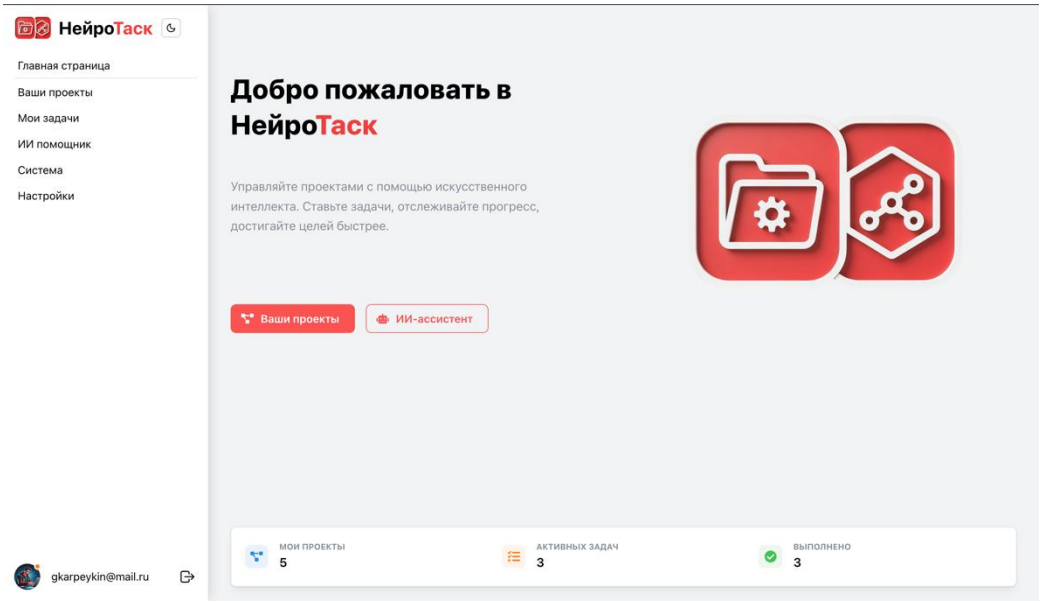


Рис. 1. Главная страница системы «НейроТаск»

Архитектура системы – микросервисная, клиент-серверная, состоящая из шести компонентов: клиентское веб-приложение, *API*-шлюз, сервис управления проектами и задачами, сервис управления пользователями и ролями, платформа автоматизации рабочих процессов *n8n* и внешняя языковая модель. Микросервисный подход обеспечивает изоляцию отказов, независимое масштабирование компонентов и возможность развития каждого сервиса без влияния на остальные [5]. Была построена схема взаимодействия компонентов (см. рис. 2).

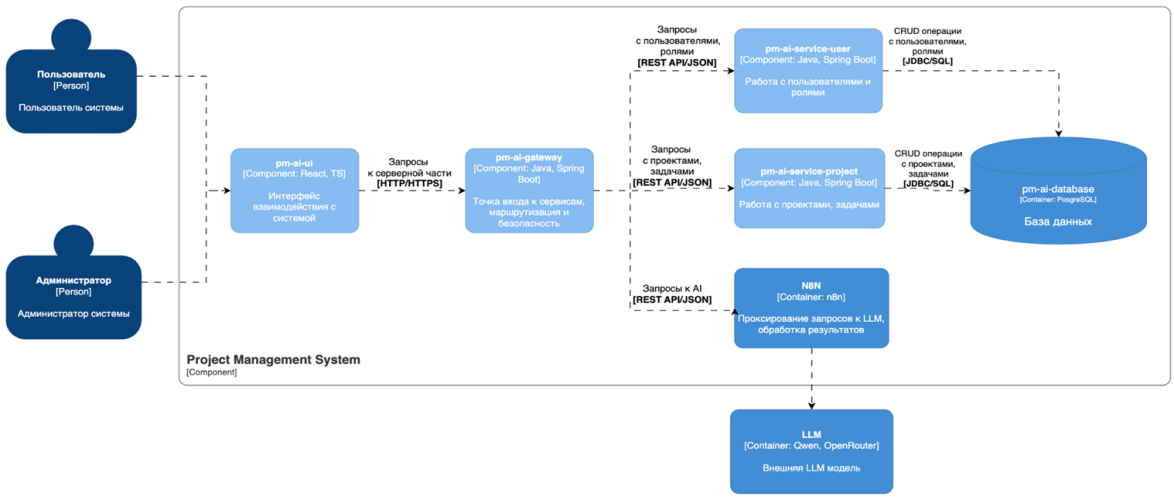


Рис. 2. Схема взаимодействия компонентов системы

В системе предусмотрены две роли: пользователь и администратор. Пользователю доступен базовый функционал управления проектами и задачами, а также ИИ-функционал – диалоговое взаимодействие с ассистентом, декомпозиция целей и генерация описания проекта. Администратору доступно управление всеми сущностями системы. Диаграмма вариантов использования приведена на рис. 3.

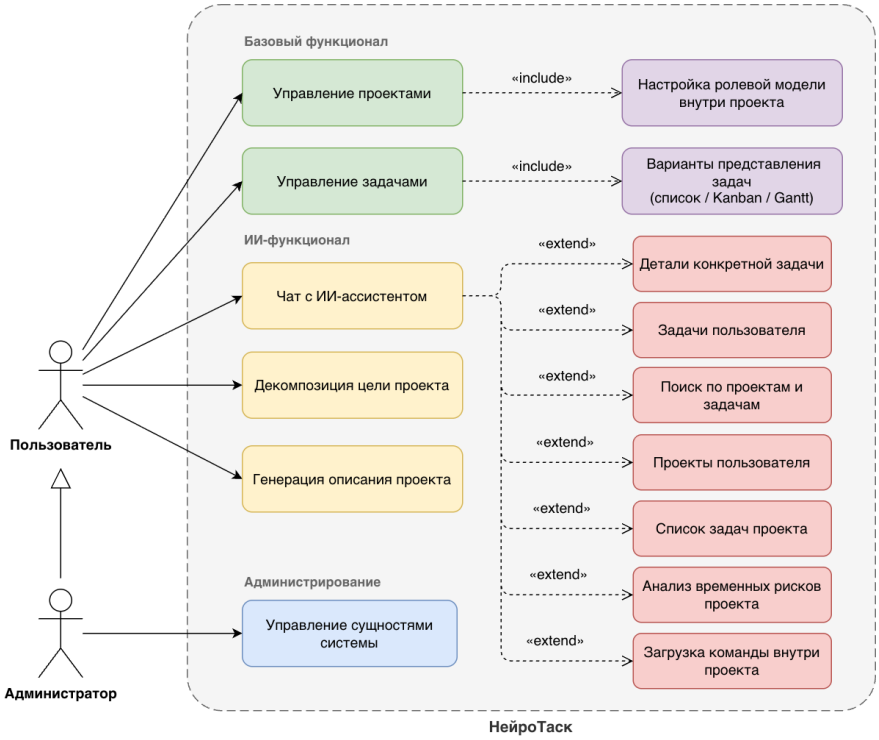


Рис. 3. Диаграмма вариантов использования системы

Ключевые функциональные блоки системы включают:

1. Управление проектами: создание, редактирование и удаление проектов; настройка статусов; добавление участников и назначение ролей. На карточке проекта отображается соответствующая информация, а также вкладки навигации (см. рис. 4).

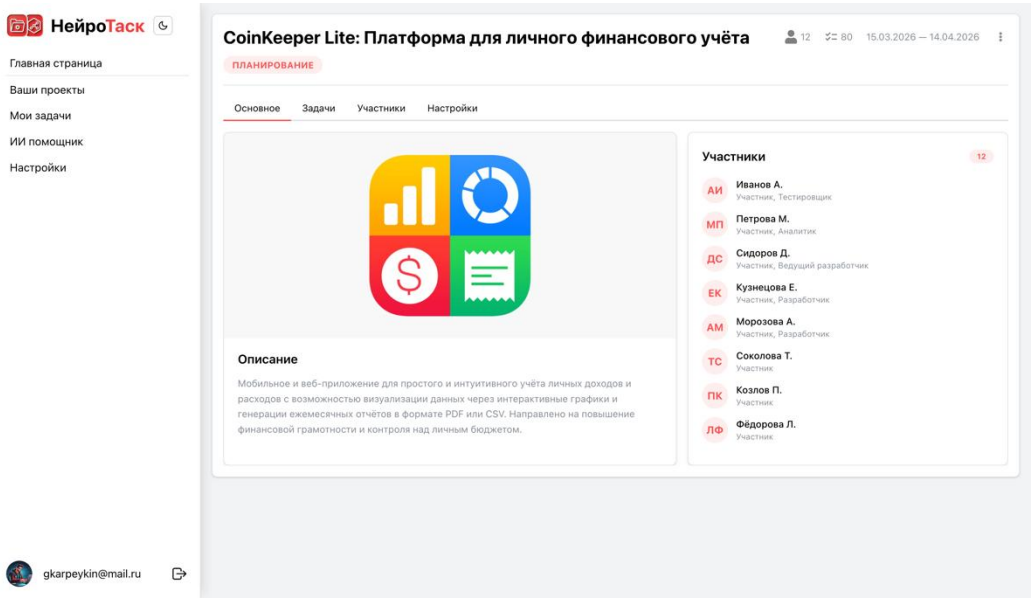


Рис. 4. Карточка проекта

2. Управление задачами: иерархическая структура задач; назначение исполнителей; установка приоритетов, сроков и статусов; межзадачные зависимости; тест-кейсы и комментарии. Визуализация задач поддерживается в трех режимах: список, *Kanban* доска и диаграмма Ганта (см. рис. 5).

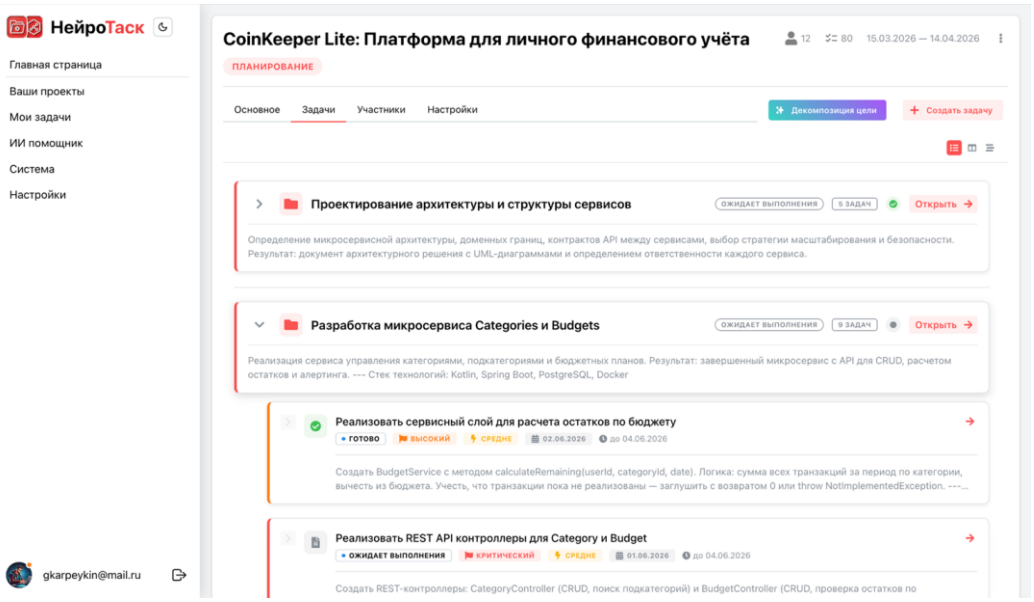


Рис. 5. Список задач проекта

3. Интеллектуальный функционал: модуль, построенный на платформе *n8n*, включает в себя как заранее определенные сценарии, так и интеллектуального ассистента, построенного на агентной архитектуре, в котором языковая модель выступает в роли интеллектуального ядра, самостоятельно принимающего решение о вызове инструментов в зависимости от запроса пользователя. Функционал обеспечивает три ключевых сценария: диалоговое взаимодействие в формате чата, автоматическую декомпозицию цели проекта с формированием иерархической структуры задач и генерацию описания проекта для автозаполнения форм [6]. При декомпозиции пользователь вводит цель, технологии и ограничения, после чего система автоматически формирует иерархическую структуру задач с предложенными приоритетами, сложностью и тест-кейсами (см. рис. 5). Результат можно отредактировать и сохранить в проект.

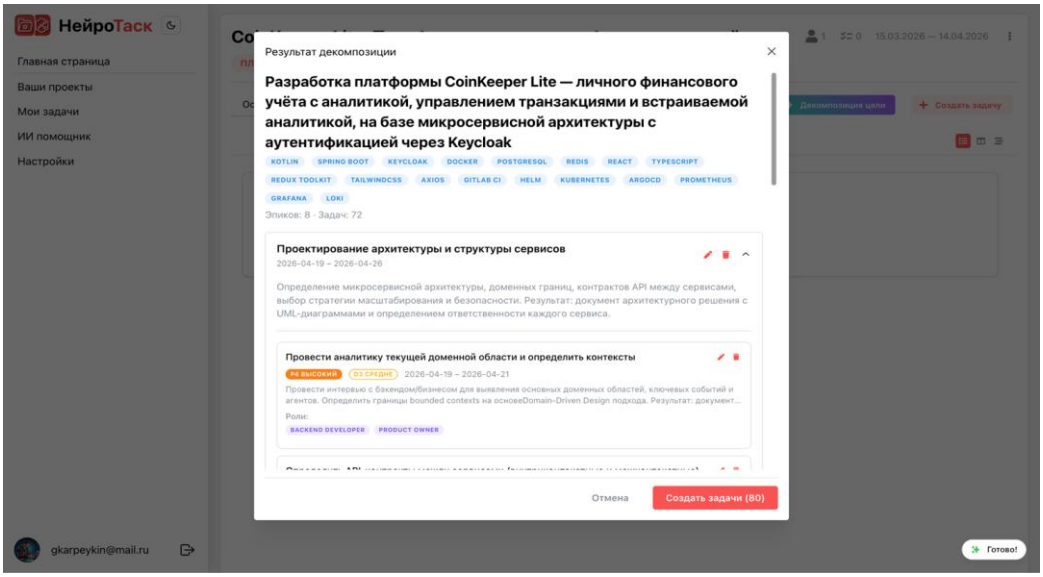


Рис. 6. Результат декомпозиции цели ИИ-ассистентом

Технологический стек системы включает: *Java* 21, *Spring Boot*, *React* 19, *TypeScript*, *PostgreSQL*, *Docker*, платформу автоматизации *n8n* и языковую модель *Qwen Coder Next* [7].

3. Искусственный интеллект в управлении проектами

Период 2025–2026 годов характеризуется стремительным внедрением технологий искусственного интеллекта в производственные процессы. Инструменты вызова функций языковыми моделями (*function calling*), протоколы взаимодействия агентов (*Agent-to-Agent*, *A2A*) и *Model Context Protocol* (*MCP*), формируют новую парадигму автоматизации интеллектуального труда [6]. Одной из наиболее перспективных областей применения этих технологий является управление проектами – дисциплина, традиционно характеризующаяся высокой трудоемкостью процессов планирования, декомпозицию целевых показателей и аналитического сопровождения.

Ключевая проблема, с которой сталкиваются малые команды и индивидуальные разработчики, заключается в отсутствии выделенного управленческого персонала [2]. В таких условиях функции планирования и аналитики вынужденно делегируются техническим специалистам. Это приводит к когнитивной перегрузке вследствие регулярного переключения контекста и снижает общую продуктивность. Интеграция ИИ-ассистента в систему управления проектами позволяет нивелировать данный дефицит. Лингвистическая модель берет на себя рутинные аналитические операции – иерархическую структуризацию целей, экспликацию описаний и первичную оценку рисков. За пользователем закрепляется функция верификации, предполагающая утверждение, корректировку или отклонение сгенерированных вариантов.

Существенным аспектом исследования является оценка влияния ИИ-ассистента на общую структуру и сложность процессов управления проектами. Внедрение интеллектуального компонента не усложняет операционную деятельность, поскольку данный функционал носит вспомогательный, а не обязательный характер. Пользователь сохраняет возможность полностью игнорировать ИИ-модуль и работать в режиме классического трекера задач. Инструменты автоматической декомпозиции и диалоговые интерфейсы локализованы в обособленных разделах системы, что исключает их влияние на базовый рабочий процесс, а сохранение за человеком функции финального утверждения минимизирует риски, связанные с генерацией некорректных данных [6].

ИИ-ассистент обеспечивает непрерывную поддержку на всех фазах жизненного цикла проекта, начиная с этапа инициации, где он содействует формулированию описания и структурированию исходного концепта. На этапе планирования алгоритмы осуществляют автоматизированное построение иерархии задач на основе заданных целей, в то время как на фазе исполнения модель генерирует оперативную аналитику без временных затрат на ручной сбор метрик. На этапе контроля система осуществляет упреждающее выявление задач, находящихся в зоне риска нарушения сроков. Подобный подход концептуально меняет роль проектного инструментария, преобразуя его из пассивной среды фиксации данных в активного участника процесса, способного к выработке рекомендаций и предупреждению о возникающих проблемах [8].

Необходимо подчеркнуть, что ИИ-ассистент не заменяет деятельность человека, а формирует структурированный проект документа, подлежащий обязательной верификации и корректировке. Полный контроль над принятием управленческих решений сохраняется за пользователем. В разработанной системе «НейроТаск» данный принцип реализован посредством механизма предварительного просмотра результатов декомпозиции и возможности их редактирования до момента интеграции в проект. Кроме того, агентная архитектура интеллектуального модуля обеспечивает высокую масштабируемость: интеграция новых функциональных сценариев и инструментов ассистента осуществляется без модификации базовых серверных компонентов системы [5].

4. Пути развития решения

Текущая реализация закладывает основу для дальнейшего развития системы по нескольким направлениям.

Перспективным направлением является переход к мультиагентной архитектуре, в рамках которой функции текущего ассистента распределяются между специализированными агентами: агент-планировщик для декомпозиции целей, агент-аналитик для мониторинга прогресса и выявления рисков, агент-документатор для генерации описаний, тест-кейсов и отчетов. Каждый агент может использовать наиболее подходящую для его задач языковую модель, а координация осуществляется

через общий контекст проекта. Такой подход повышает качество результатов в каждой из зон ответственности и позволяет более гибко наращивать функциональность системы [6].

Существенный потенциал для повышения производительности и автономности системы связан с переходом на локально развертываемые языковые модели. Использование открытых моделей, оптимизированных для выполнения на пользовательском оборудовании, позволит устранить задержки, связанные с обращением к внешним *API*, обеспечить конфиденциальность проектных данных и снизить эксплуатационные расходы [6].

Расширение функциональных возможностей может быть достигнуто за счет интеграции с внешними сервисами. Интеграция с системами контроля версий (*GitHub*, *GitLab*) позволит связать задачи с соответствующими ветками и коммитами, автоматизируя отслеживание прогресса. Интеграция с конвейерами *CI*/*CD* обеспечит отображение статуса сборок и тестов непосредственно в интерфейсе задачи, позволяя команде оперативно реагировать на регрессии. Интеграция с мессенджерами расширит коммуникационные возможности: уведомления о событиях проекта и создание задач непосредственно из интерфейса мессенджера.

При переходе к промышленной эксплуатации целесообразна замена *Docker Compose* на оркестрацию *Kubernetes*, обеспечивающую автоматическое масштабирование микросервисов под нагрузкой, автоматический перезапуск отказавших компонентов и управление конфигурацией через стандартизированные манифесты [5].

Заключение

В рамках работы спроектирована и реализована информационная система управления проектами «НейроТаск» с интегрированным ИИ-ассистентом, ориентированная на малые команды и индивидуальных разработчиков. Проведено исследование предметной области и выполнен сравнительный анализ существующих решений – *Jira*, *Trello*, *YouGile*, *ClickUp*, – по результатам которого выявлена незаполненная ниша: инструмент, сочетающий простоту освоения с бесплатным ИИ-функционалом [4]. Спроектирована микросервисная архитектура на основе шести компонентов, обеспечивающая независимое масштабирование и отказоустойчивость. Реализован прототип, включающий клиентское веб-приложение и серверные микросервисы с интеграцией языковой модели через платформу автоматизации *n*8*n*.

Практическая значимость работы определяется тем, что разработанная система снижает порог входа в проектное управление для малых команд за счет автоматизации рутинных аналитических задач – декомпозиции целей, генерации описаний и анализа состояния проектов. ИИ-ассистент, построенный на агентной архитектуре, обеспечивает интеллектуальную поддержку на всех этапах жизненного цикла проекта, не заменяя пользователя, а предоставляя основу для принятия решений [2].

Перспективными направлениями дальнейшего развития являются переход к мультиагентной архитектуре, локальное развертывание языковых моделей, расширение функциональных возможностей за счет интеграции с системами контроля версий, *CI*/*CD* и мессенджерами, а также переход на оркестрацию *Kubernetes* для промышленной эксплуатации.

Список источников

1. Аламмар, Дж. Большие языковые модели на практике: Понимание языка и генерация текстов / Дж. Аламмар, М. Гроотендорст. – Астана : Алист, 2026. – 416 с.
2. Григорьев М. В. Проектирование информационных систем: учебное пособие для вузов / М. В. Григорьев, И. И. Григорьева. – Москва: Юрайт, 2021. – 318 с.
3. Пачеко, Д. Принципы модернизации программных архитектур: построение архитектур на основе микросервисов, монолитов и распределенных монолитов / Д. Пачеко, С. Сгро. – Санкт-Петербург: БХВ -Петербург, 2026. – 496 с.
4. Пименов, А. А. Канбан Метод. Базовая практика / А. А. Пименов. – Санкт-Петербург: Питер, 2025. – 208 с.

5. Таулли, Т. Программирование с помощью искусственного интеллекта : Улучшенное планирование, кодирование, тестирование и развертывание / Т. Таулли. – Астана: АЛИСТ, 2025. – 240 с.

6. Турчак, В. Импортозамещение Trello, Jira и Confluence – подборка для разработчиков // TProger: [сайт]. – Дата публикации: 16.10.2025. – URL: https://tproger.ru/articles/importozameshhenie-trello--jira-i-confluence---podborka-dlya-razrabotchikov.

7. Урванов, Ф. В. Spring и Spring Boot. Разработка облачных приложений на Java / Ф. В. Урванов. – Санкт-Петербург: БХВ, 2025. – 384 с.

8. Чекмарев, А. В. Управление цифровыми проектами и процессами : учебник для вузов / А. В. Чекмарев. — 2-е изд., перераб. и доп. — Москва : Юрайт, 2026. — 424 с.