

СРАВНИТЕЛЬНЫЙ АНАЛИЗ JIRA, ASANA, TRELLO, MICROSOFT AZURE DEVOPS, SCRUMWORKS, VERSIONONE, GITLAB И GITHUB В СИСТЕМЕ НОРМИРОВАНИЯ ТРУДА

Смолина Е.Р., магистрант группы СГНЗ-41М

Московский государственный технический университет им Н. Э. Баумана

Научный руководитель: Гришнова Е.Е., доктор политических наук, профессор кафедры «Информационная аналитика и политические технологии»

Аннотация. В статье рассматриваются цифровые системы управления задачами как информационная основа организации и нормирования труда. Цель исследования состоит в выявлении общих и особенных свойств JIRA, Asana, Trello, Microsoft Azure DevOps, ScrumWorks, VersionOne, GitLab и GitHub, а также в определении задач, для которых каждое решение наиболее уместно. Исследование основано на сравнительном анализе научных публикаций и официальных описаний функциональных возможностей данных приложений.

Ключевые слова: организация труда, нормирование труда, управление задачами, цифровизация труда, проектное управление, прослеживаемость работ, трудоемкость.

Цифровизация организационных процессов усилила значение информационных систем, которые фиксируют не только сам факт постановки задания, но и движение работы между исполнителями, длительность цикла, причины задержек и результат приемки. В этой связи системы управления задачами и совместной работой становятся для нормирования труда частью информационной инфраструктуры организации. Именно они позволяют переводить деятельность команды из неформального обмена поручениями в сопоставимую и наблюдаемую форму.

При этом такие системы не устанавливают нормы автоматически. Их ценность состоит в другом: они разделяют труд сотрудников на единицы учета – задачи, дефекты, запросы, крупные блоки работ, элементы перечня продукта, и тем самым создают основу для сравнения фактической и ожидаемой трудоемкости. Цель статьи заключается в выявлении общего и особенного в рассматриваемых инструментах и в определении того, для каких задач и типов команд они подходят в наибольшей степени.

Для анализа труда существенны не названия функций, а те данные, которые система позволяет извлечь из повседневной работы. В прикладном отношении особенно важны пять групп показателей: объем работ, трудоемкость, длительность цикла, устойчивость потока и качество результата. Если цифровая система фиксирует исполнителя, сроки, изменение состояний, зависимости, оценки, фактические затраты времени, причины блокировки и основания завершения, она становится пригод-

ной для пересмотра норм и распределения нагрузки на более строгой основе [1].

Однако сама по себе цифровая фиксация еще не гарантирует корректности выводов. Сопоставимость появляется только тогда, когда в организации единообразно определены типы работ, критерии начала и завершения задачи, основания возврата и правила приемки результата. Иначе одинаковые статусы в разных командах будут описывать разные трудовые состояния и исказить нормативный анализ [2].

Общим для всех рассматриваемых систем является наличие базового контура организации совместной работы: разбиение деятельности на задачи, закрепление ответственного, указание срока, визуализация состояния исполнения и поддержка коллективного обсуждения. Именно поэтому любая из этих систем способна повысить прозрачность труда на базовом уровне.

Различия сосредоточены в четырех параметрах: глубина настройки процессов, уровень управленческого обзора, степень связи задачи с изменением продукта и развитость аналитики. Сравнительные исследования показывают, что более функционально насыщенные инструменты лучше поддерживают координацию средних и крупных проектов, но при чрезмерной сложности могут снижать удобство использования и увеличивать учетную нагрузку [6–8].

Рассмотрим детальнее группы инструментов и предпочтительные сценарии их применения. Asana и Trello представляют универсальный контур организации работы. Asana уместнее там, где требуется координация нескольких подразделений, календарное представление процесса, обзор портфеля задач и контроль загрузки. По этой причине она особенно подходит для проектной, административной, маркетинговой и иной межфункциональной деятельности. Trello, напротив, рационален в небольших командах и простых потоках работ, когда важнее быстро получить наглядную доску задач, видимость очередности процессов и обеспечить дисциплину прохождения стадий проектов без сложной настройки и особых навыков. В сравнении с Asana, JIRA или Azure DevOps его возможности аналитики и многоуровневой координации заметно уже, что делает Trello сильным прежде всего как инструмент простого визуального управления.

JIRA, Microsoft Azure DevOps, VersionOne и ScrumWorks ближе к формализованной координации инженерной или программной работы. JIRA удобна при большом числе типов задач, ролей, зависимостей и маршрутов согласования; ее сильная сторона состоит в настраиваемых процессах и отчетности. Azure Boards оправдан там, где планирование работ, учет дефектов и командная координация должны быть тесно связаны с технологическим контуром Microsoft. VersionOne в современной

корпоративной поставке фактически продолжает линию Digital.ai Agility и потому более уместен в крупных программах, где важны портфельный обзор, приоритизация и несколько уровней планирования. ScrumWorks имеет более узкую специализацию и целесообразен прежде всего в командах, работающих с гибкими методологиями и спринтами, где не требуется широкий корпоративный контур управления.

GitLab и GitHub выделяются тем, что связывают учет работы с реальным изменением продукта. GitLab объединяет требования, задачи, крупные элементы планирования, контроль сроков, доски, дорожные карты и учет времени, а затем позволяет продолжить процесс в единой среде проверки и выпуска результата. Идеально подходит для работы с программным кодом. Git Hub Projects также поддерживает задачи, запросы на изменение, табличные, досочные и дорожные представления, а также настраиваемые поля, но сознательно сохраняет более гибкий и менее жестко регламентированный подход. Поэтому GitLab предпочтителен, когда нужна сквозная прослеживаемость от постановки задачи до проверенного результата, тогда как GitHub особенно эффективен в распределенных командах, для которых центральной остается совместная работа с репозиторием и проверкой внесенных изменений коллегами.

Практическая ценность рассмотренных систем состоит в том, что они делают труд наглядным и контролируемым. Руководитель видит не только итог, но и структуру движения работы: где формируется очередь, какие категории задач задерживаются, насколько фактическая длительность работ отклоняется от плановой. Для повторяющихся категорий деятельности это создает основу для уточнения типовых норм и более точного распределения нагрузки.

Однако цифровой контур учета имеет и ограничения. Так, формально закрытая задача может не совпадать с реально завершенной работой, избыточное число статусов и полей может существенно увеличивать непроизводительные затраты времени [3]. Необходимо также учитывать, что исследовательскую и творческую работу труднее формализовать в таких системах. Следовательно, инструмент начинает мешать организации труда, если учет превращается в самоцель и перестает обслуживать реальный процесс.

Сопоставление инструментов показывает, что их выбор должен определяться типом работы и требуемой глубиной управленческого учета. В обобщенном виде предпочтительные области применения можно представить следующим образом (Таблица 1).

Таким образом, Trello и Asana рационально использовать там, где главная задача состоит в упорядочивании организационного труда и координации между подразделениями. JIRA и AzureDevOps предпочтительны для сложной инженерной среды, требующей детальной форма-

лизации и устойчивой прослеживаемости. VersionOne / Digital.aiAgility оправдан там, где объектом управления становится не отдельная команда, а программа или портфель. ScrumWorks полезен для сравнительно узкого класса команд, строго работающих по Scrum. GitLab и GitHub оптимальны в тех случаях, когда учет труда должен быть неразрывно связан с фактическим изменением продукта.

Таблица 1

Предпочтительная область применения систем

Система	Сфера применения	Основные достоинства	Основные ограничения
Trello	небольшие команды, простые линейные потоки задач	быстрый запуск, наглядность, низкий порог освоения	ограниченная глубина аналитики и слабая многоуровневая координация
Asana	межфункциональные проекты, административные и проектные процессы	календарная логика, формы, правила, отчетность, управление загрузкой	труднее выстроить связь с кодом, проверкой качества и выпуском результата
JIRA	сложные продуктовые и инженерные процессы с большим числом ролей и состояний	гибкие маршруты задач, детальная структуризация, доски и ограничения незавершенной работы	сложность настройки и риск избыточного учета
Azure DevOps	инженерная среда, особенно в технологическом стеке Microsoft	сквозная прослеживаемость, связь задачи с репозиторием, тестированием и выпуском	избыточность для чисто административных процессов
VersionOne / Digital.ai Agility	программы и портфели с несколькими командами и уровнями планирования	дорожные карты, метрики, межкомандная координация, портфельный обзор	требовательность к зрелости процесса и качеству данных
ScrumWorks	Команды, работающие с гибкими методологиями по спринтам	простота Scrum-ориентированного учета, визуализация хода итерации	узкая специализация, отсутствие экосистемы
GitLab	разработка, где задача должна быть доведена до выпуска изменения	аналитика потока, измерение стадий, связь задачи, кода, проверки и выпуска	менее удобен для нетехнических подразделений
GitHub	распределенная разработка, где важны задачи, обсуждение и проверка изменений	гибкие представления проектов, настраиваемые поля, естественная связь с запросами на изменение	менее жесткое портфельное управление по сравнению с корпоративными системами

Важно отметить, что независимо от выбранного продукта, для целей нормирования труда необходимы единый классификатор видов работ, сопоставимая крупность задач и обязательная фиксация исполнителя, срока, состояния, причины блокировки и основания завершения. Кроме того, производительные, вспомогательные и срочные работы следует анализировать отдельно, чтобы нормативная база отражала устойчивую структуру труда, а не случайные отклонения процесса.

Подводя итог, проведенный анализ показывает, что универсального лидера среди JIRA, Asana, Trello, Microsoft Azure DevOps, ScrumWorks, VersionOne, GitLab и GitHub не существует. Их общее значение заключается в повышении наблюдаемости и сопоставимости труда, а различия определяются степенью формализации процесса, уровнем управленческого обзора и глубиной связи задачи с изменением продукта. Следовательно, корректный выбор системы должен определяться типом труда, зрелостью регламентов и требуемой глубиной аналитики.

Литература и источники:

1. Ануфриева И.Ю. Трансформация внутриорганизационного нормирования труда в условиях цифровизации // Экономика. Профессия. Бизнес. 2021. № 3. С. 12–18.
2. Доманский В.О., Тарханова О. В., Пелевин М. Д. Анализ возможностей мобильных Agile-решений для эффективной проектной деятельности // Архитектура, строительство, транспорт. 2021. № 3. С. 98–105.
3. Подвербных О.Е., Самохвалова С.М., Донова Н.А., Мокин Ю.И. Формирование и реализация модели цифровой организации труда на промышленном предприятии // Креативная экономика. 2023. Т. 17. № 12. С. 4945–4964.
4. Противодействие фальсификации истории великой отечественной войны / Бочарников И.В., Суздалева Т.Р., Федоров К.В., Криворучко А.А., Петренко А.И., Зеленков М.Ю., Кандыбович С.Л., Разина Т.В., Овсянникова О.А., Трипольский В.Б. Москва, 2020.
5. Ремарчук В.Н. Информационно-аналитическая деятельность: проблемы и перспективы // Вестник Академии военных наук. 2023. № 1 (82). С. 31–35.
6. Юсупова С.М. Регламентация труда в российских организациях в условиях цифровой экономики // Гуманитарный научный журнал. 2022. № 2. С. 3–14.
7. Mishra A., Mishra D. Software Project Management Tools: A Brief Comparative View // ACM SIGSOFT Software Engineering Notes. 2013. Vol. 38. No. 3. P. 1–4.
8. Stettina C.J., Horz J. Agile portfolio management: An empirical perspective on the practice in use // International Journal of Project Management. 2015. Vol. 33, No. 1. P. 140–152.