

УДК 004.42:005.83
DOI: 10.15827/0236-235X.139.374-383

Дата подачи статьи: 25.07.22, после доработки: 05.08.22
2022. Т. 35. № 3. С. 374–383

Методика упрощения таск-трекинга в проектах игровой индустрии

*Р.А. Шараева*¹, младший научный сотрудник, *r.sharaeva3496@gmail.com*

*В.В. Кугуракова*¹, к.т.н., доцент, руководитель лаборатории,
vlada.kugurakova@gmail.com

*Н.Э. Селезнева*¹, аспирант, *nataliselezneva306@gmail.com*

¹ Казанский (Приволжский) федеральный университет,
Лаборатория разработки интеллектуальных инструментов компьютерных игр,
Институт информационных технологий и интеллектуальных систем,
г. Казань, 420008, Россия

Разработка компьютерных игр отличается от традиционной разработки проектов в сфере информационных технологий, главным образом, высокой степенью креативности процессов, что необходимо учитывать при применении методов программной инженерии.

Темой настоящего исследования стали изучение особенностей автоматизации процессов управления проектами, а также разработка инструмента для управления задачами (таск-трекинг), воплощающего методики и рекомендации, сформированные для игровой индустрии. Для качественных формулировки цели и постановки задач проанализированы актуальные научные источники, в результате выделен ряд функций, необходимых для командных таск-трекеров: присвоение сотруднику роли или задачи, создание зависимых задач, диаграмма Ганта, доступ сотрудника к информации о связанных задачах, распространение ссылок и документов, создание повторно используемых шаблонов.

По результатам проведенного исследования представлена методика, которая упрощает работу менеджера проекта с точки зрения ведения задач. Разработанный подход предполагает создание следующих возможностей: автоматическое распределение задач исполнителям (в соответствии с направлением их деятельности), автоматическое создание дубликатов и настройка зависимости задач, автоматическая передача артефактов зависимым задачам, генерация шаблона проекта по брифу – опросному листу с ключевыми аспектами проекта. На примере управления процессом разработки приложения с дополненной реальностью было показано, что полученные результаты актуальны для разработки интерактивных, геймифицированных проектов. Для проверки эффективности этих результатов сформированные практические приемы были интегрированы в систему с открытым исходным кодом (LeanTime).

Предложенный подход показал сокращение времени решения управленческих задач более, чем на 20 %, а в случае потребности в множестве новых шаблонов сокращение может превысить 40 %. Созданный веб-инструмент можно использовать в качестве инструмента автоматизации процессов разработки в любых узкоспециализированных ИТ-направлениях.

Ключевые слова: *программная инженерия, ИТ-разработка, игровая разработка, таск-трекер, автоматизация.*

По данным аналитики маркетплейса My.Games, за 2021 год российская индустрия видеоигр по объему вложенных финансовых ресурсов выросла до 177,4 млрд рублей. Помимо развлекательных целей, компьютерные игры могут внести большой вклад в освоение новых навыков благодаря геймификации процесса обучения. Сформировалось понятие «серьезные игры», которые стали активно развиваться в последнее время, погружаясь во многие сферы нашей жизни, включая здравоохранение, маркетинг и бизнес [1, 2]. Геймификация – это внедрение игровых форм в неигровой контекст: работу, учебу и повседневную жизнь.

Несмотря на то, что создание видеоигр вообще и разработка разнообразных геймифицированных проектов (геймдев-проектов) относятся к направлению программной инженерии, выделяют несколько аспектов, отличающих эту разработку от разработки ПО в общем случае. Прежде всего, сфера разработки видеоигр тесно связана с креативом и творческим процессом. Такая разработка требует участия, помимо программистов, разнообразных специалистов, например, сценаристов, музыкантов, художников. Игровые проекты чаще реализуются по гибким методологиям, когда предъявляемые требования менее четкие, чем для классических программных продуктов [3]. С уче-

том перечисленных аспектов необходим особый подход к таск-трекингу таких проектов.

Базовый функционал по управлению проектами разработки ПО – в общем случае давно решенная задача, но остается актуальным расширение возможностей используемого инструментария на специализированные сферы разработки.

Связанные работы

С точки зрения управления решением задач в сфере разработки ПО облачные технологии позволили менеджерам следить за процессами, реализуемыми в выполняемом проекте, дистанционно и с большой точностью. Интегрированность онлайн-технологий в ИТ-проекты максимально проявилась и в веб-сервисах, благодаря чему распространенными являются такие инструменты, как Trello (<https://trello.com/ru>) и даже более профессиональная система Asana (<https://asana.com/>).

С научной точки зрения интересны как эффективность, так и перспективы развития таск-трекеров, а развитие автоматизации в этой сфере позволяет сократить десятки часов работы менеджеров. В свою очередь, это дает возможность распараллелить производство проектов, а также упрощает ведение менеджерами нескольких проектов одновременно. Автоматизация рутины процесса управления проектами также может снизить порог вхождения в профессию, так как часть ошибок начинающих специалистов может быть нивелирована автоматизированными алгоритмами.

В [4] проанализированы 60 таск-трекеров для совместной работы с перечнем критериев оценки инструментов. Среди них интересны следующие функции: планирование на основе взаимосвязанных задач, создание диаграммы Ганта, возможность делиться ссылками, файлами и другими данными.

Существует ряд работ, которые затрагивают тему менеджмента в конкретной отрасли. Например, разработчики веб-трекера WebTP сформулировали требования к коллаборативному инструменту ведения веб-проекта [5].

Свое развитие получили как проприетарные инструменты для управления процессом разработки ПО, так и системы с открытым исходным кодом. Открытые системы по реализованному функционалу ничуть не отстают от популярных проприетарных, однако акцент на удобство использования больше присущ второму типу инструментов [6].

Актуальными для таск-трекера остаются следующие функции: присвоение сотруднику роли или задачи, создание взаимозависимых задач, диаграмма Ганта, доступ сотрудника к информации о связанных задачах, распространение ссылок и документов, создание повторно используемых шаблонов.

В 2018 году был проведен анализ трех зарубежных инструментов управления проектами: Asana, Freedcamp и Ace Project [7]. Подсвеченные отсутствующие функции Asana, такие как создание зависимостей между задачами, установка приоритетов, отсутствие диаграммы Ганта, были внедрены в более поздних обновлениях системы в бесплатной, премиум- или бизнес-версиях.

Поиск подходов для повышения эффективности инструментов для автоматизации проектного менеджмента продолжается, разрабатываются расширения существующих инструментов. Так, для Trello был выработан плагин, позволяющий отслеживать эмоциональное состояние сотрудников при получении технических задач [8].

Сбор статистических данных об активности участников проектной команды и их дальнейший анализ также вызывают интерес у руководителей проекта. Автоматизация отслеживания снимков кода и данных о действиях разработчиков предлагают в инструменте TaskTracker-tool [9].

В ходе анализа имеющихся источников авторами данного исследования не было найдено специализированное решение для автоматизации управленческих задач в сфере разработки игр. Однако получено достаточно информации для разработки целостной и эффективной концепции создания инструмента управления задачами и проектом в целом.

Существующие примеры автоматизации таск-трекинга в геймдев. По мнению интернет-издания Game Career Guide [10], в сфере разработки видеоигр наиболее популярными являются следующие системы таск-менеджмента.

- Asana – универсальная система управления проектами, разработанная в США. Имеет множество интеграций с другими инструментами, например, для учета времени или хранения и передачи артефактов разработки.
- Monday (<https://monday.com/lang/ru>) – универсальная система проектного менеджмента из Израиля. Предоставляет готовые шаблоны проектов без возможности заготовки задач. Передача файлов между участниками

команды доступна посредством интеграции. Решение предлагает настройку автоматизации через задание правил, например, для назначения конкретного исполнителя [11].

- Trello – австралийская универсальная система для ведения проектов. Инструмент предлагает гибкую настройку автоматизации с учетом правил [12], а также готовые шаблоны проектов.

- HacknPlan (<https://hacknplan.com/>) – система, специально предназначенная для ведения проектов из геймдев-сферы и разработанная в Испании. Особенностью инструмента является автоматическая генерация GDD (полного описания разрабатываемого игрового проекта). В решении отсутствует возможность настройки автоматизации и не реализована функция передачи результатов работ.

- Google Drive (https://www.google.com/intl/ru_ru/drive/) – хранилище файлов, разработанное американской компанией Google. Система не предназначена для полноценного таск-трекинга и управления проектом. Целевой функцией решения является возможность обмена данными с передачей дополнительных комментариев.

- Basecamp (<https://basecamp.com/>) – американское универсальное решение для управления проектами, предоставляющее функционал для создания проектов с использованием заранее заготовленных шаблонов. В системе недоступна настройка автоматизации и отсутствуют встроенный функционал и интеграция для передачи файлов с дополнительной информацией для корректной интеграции.

- Assembla (<https://get.assembla.com/>) – универсальная система управления проектно-ориентированным бизнесом, которая создана специально для команд, занимающихся разработкой ПО. Создатели решения делают акцент на функциях контроля версий, автоматическом резервном копировании и на безопасности продуктов пользователей. Несмотря на это, в инструменте отсутствует возможность настраиваемой автоматизации дубликатов задач и назначения исполнителей.

- Jira (<http://jira.ru/>) – популярный инструмент для управления проектами из США, предлагающий готовые шаблоны без заготовки задач, передачу файлов между участниками команды посредством интеграции и автоматизацию назначения исполнителей, а также других настраиваемых правил [13].

Настраиваемая автоматизация в существующих системах предусматривает настройку

правил для автоматизации, например, назначение конкретного или свободного исполнителя при смене статуса задачи. Однако она не учитывает проектную роль участников команды.

В системе управления проектами Asana [14] есть возможность автоматического создания дубликата задачи при назначении дополнительного исполнителя. Такой подход может позволить значительно сократить время, но он не учитывает последовательность выполнения задач.

Только в двух решениях (Asana и HacknPlan) доступно создание шаблонов: у пользователя есть возможность формирования проекта с заготовленными задачами и дальнейшего использования его в качестве шаблона [15, 16]. Однако в сфере разработки геймдев присутствуют множества библиотек и различных механик [17], что при наличии всевозможных шаблонов, учитывающих все эти комбинации, не ускорит процесс старта проекта, а замедлит его из-за долгого выбора подходящего шаблона. Поэтому требуется пересмотр парадигмы шаблонизации.

Передача файлов, встроенная или организованная путем интеграции, в выявленных авторах статьи системах представляет собой непосредственно отсылку или передачу доступа. Но при специфике геймдев-проектов и необходимости разбиения команды их исполнителей на группы по имеющимся навыкам деятельности необходима передача дополнительных данных для интеграции в проект артефактов. Напрашивается аналогичный вывод – требуется уточнение процедуры передачи материалов от одного участника команды другому.

Общее заключение: необходимо пересмотреть все описанные механики автоматизации, привычно используемые в таск-трекерах в геймдев. При этом система Asana является ключевым аналогом решения, разрабатываемого в рамках настоящего исследования, так как включает в себя и автоматизацию назначения и дублирования задач, и создание шаблонов с заготовкой задач.

Особенности разработки геймдев-проектов. Многофункциональная команда разработки. Выделяют две наиболее популярные методологии разработки геймдев-проектов: водопадную (Waterfall) и гибкую (Agile) [2]. Учитывая риски, водопадную разработку ПО стоит применять лишь при условии полного понимания и неизменности требований, а также в случае «штамповки» подобных решений. Методо-

логия Agile является более подходящей для ведения проектов по созданию игровых приложений и приложений с дополненной реальностью, так как в ней большее внимание уделяется качеству ПО, пожеланиям заинтересованных лиц и быстрому получению рабочего прототипа. Однако стоит учесть, что для сферы разработки видеоигр (<https://vc.ru/hr/174739-komandy-gazrabotchikov-sostav-i-rol-i-v-gamedev>) характерна многофункциональная команда разработки, включающая как классические (проектный менеджер/scrum master, аналитик, архитектор/техлид, UI/UX-дизайнер, программист/разработчик, тестировщик/QA, заказчик/владелец продукта/producer), так и узкоспециализированные роли (например, арт-лид, нарративный дизайнер; сценарист, level-дизайнер/дизайнер уровней, 3D-дизайнер, аниматор, 2D-дизайнер, саунд-дизайнер/композитор). Поэтому существуют гибкие методологии, учитывающие аспекты и проблемы геймдев-сферы [18]. Для целей конкретного проекта список специальностей, конечно, может быть расширен. Главный вывод – задачи не могут быть назначены любому свободному участнику команды, а должны быть выполнены только одним из целевых специалистов, владеющим определенными компетенциями. В командах разработки обычного ПО таких узких специальностей намного меньше.

Ответы на шаблонные вопросы формируют рутинные. В зависимости от выбранного фокуса геймдев-приложения необходим стартовый набор ответов на вопросы (будем называть его «бриф»), которые влияют на реализацию проекта (например, если речь идет об AR-приложении: На каком устройстве воспроизводится приложение с дополненной реальностью? Какой игровой движок используется при разработке? Какая платформа дополненной реальности используется? Какой маркер используется для запуска дополненной реальности? Взаимодействует ли пользователь с элементами дополненной реальности? Нужен ли пользовательский интерфейс? Какой тип контента воспроизводится? Нужен ли сценарий?). Список вопросов может быть расширен в сторону более детального разбиения, а также путем введения дополнительных критериев. Ответы на эти вопросы определяют, какими навыками должны обладать специалисты в команде, и позволяют генерировать рутинные процедуры, необходимые на старте проекта.

Сильно связанный процесс разработки и слабые коммуникации. Проектный менеджмент

разработки игры, как и других видов ПО, включает постановку и распределение задач, управление и контроль их выполнения [2]. По существующим сложностям сфера разработки игр незначительно отличается от традиционной разработки, имея общие с последней актуальные проблемы: отставание от графика, возникающее в связи с неправильной оценкой и неучетом рисков, или неверная оценка бюджета и масштабов проекта [19]. Помимо наличия большого количества креативных задач, есть риск упустить в оценке проекта базовые задачи, которые так или иначе должны быть решены, а также последовательность выполнения задач. В команде с участниками, каждый из которых имеет узкоспециализированные навыки, решаемые задачи по умолчанию должны быть привязаны к конкретным исполнителям в зависимости от их специализации, однако такое четкое разделение порождает проблему отсутствия коммуникации между группами разработки [18]. Вторым сложным моментом при такой многофункциональной команде является простаивание специалиста одного направления, ожидающего, когда решит задачу участник с другим навыком [19], – в противовес, например, веб-разработке, когда frontend- и backend-части могут быть реализованы параллельно.

Авторам не известны подходы к реализации системы управления проектами, которая могла бы позволить решить проблемы, описанные выше.

Предлагаемая методика

После детального анализа особенностей геймдев-проектов был выработан подход к созданию системы таск-трекинга, позволяющей упростить управление задачами их реализации.

1. *Исполнители автоматически получают задачи по своему направлению.* При этом в зависимости от размера проекта исполнителей для заданного направления один или несколько участников команды могут иметь множество ролей. Реализовать такое автоматическое назначение задач можно, опираясь на метки к ней. Назначение меток должно модерироваться через справочник в панели администратора (например, при указании на задаче метки 3D-модель она будет назначена исполнителю с ролью 3D-дизайнер). Если претендентом на решение задачи является не один член команды, задача будет назначена лидеру группы разработки (арт-лиду – для задач визуальной части,

техлиду – для программистов) для дальнейшего выбора исполнителя в подчиненной ему группе разработки уже вручную.

2. *Автоматическое дублирование и настройка зависимостей между задачами.* В рассматриваемой сфере выделяют креативные задачи и задачи по программированию, при этом большинство требований связано с разработкой контента и дальнейшей интеграцией [20]. Описание и иная информация, например, референсные данные, сроки исполнения, могут быть едиными для всех задач для реализации требования. При такой специфике можно сократить время на дублирование и распределение задач. В качестве примера приведем простейший сценарий AR-приложений: по распознаванию маркера дополненной реальности с помощью камеры смартфона на экране должен появиться скачущий мяч. Какие же задачи необходимо осуществить для выполнения требования? Минимальный их набор: создание трехмерной модели мяча (роль исполнителя – 3D-дизайнер), анимирование мяча (роль исполнителя – аниматор), интеграция мяча в проект (роль исполнителя – программист), сборка проекта (роль исполнителя – программист). Необходимо учесть и порядок выполнения задач: 1 – создание модели, 2 – анимация, 3 – интеграция модели, 4 – финальная сборка проекта.

Исходную идею назначения исполнителей логично расширить до алгоритма автоматизации, представленного на кратких блок-схемах (рис. 1–3), более детальные блок-схемы представлены соответственно в приложениях (см. <http://www.swsys.ru/uploaded/image/2022-3/2022-3-dop/28.jpg>, <http://www.swsys.ru/uploaded/image/2022-3/2022-3-dop/29.jpg>, <http://www.swsys.ru/uploaded/image/2022-3/2022-3-dop/30.jpg>).

Если невозможно полностью описать общее требование для дубликатов задачи, возможны корректировка описания дубликатов задачи, а также добавление дополнительной информации для реализации.

К подходу, описанному выше, можно отнести и такую стадию выполнения задачи, как проверка лидерами групп разработки технической реализации, а качества – тестировщиками, путем добавления соответствующих меток и учета последовательности выполнения. Конечно, для этого должна быть настроена система меток.

3. *Автоматическая передача артефактов зависимым задачам.* Для решения проблемы коммуникации в команде разработки и пере-



Рис. 1. Создание дубликатов задачи

Fig. 1. Creating task duplicates



Рис. 2. Настройка зависимости между задачами по меткам

Fig. 2. Setting the dependency between tasks by labels

дачи артефактов между исполнителями необходим функционал фиксации результата выполнения задач. У исполнителя задачи должна быть возможность как сохранить сам артефакт разработки, так и оставить дополнительные комментарии. Для корректной работы процесса менеджеру проекта или scrum-мастеру необходимо обеспечить добросовестную фиксацию результатов по итогам завершения выполнения задачи исполнителем. По выполнении одной задачи ее результат дублируется зависящим задачам. При этом отображение сохраненных артефактов для всей цепочки зависимых задач позволит снизить риск пропуска важных аспектов артефактов разработки



и их модификаций для дальнейшей работы. Важно отобразить цепочку событий в иерархическом виде для лучшей визуализации процесса разработки со ссылками на детали задач-предшественников. Если переданной информации задаче-наследнику будет недостаточно, имея ссылку на предшествующую задачу и ее исполнителя, можно инициировать дополнительное обсуждение.

4. *Генерация шаблона проекта по брифу.* Бриф представляет собой опросный лист с ключевыми инструментами, механиками и иными аспектами проекта. Заготовка задач и заметок по ответу на эти вопросы позволяет сгенерировать начальное представление проекта в таск-трекере, минуя рутинные этапы инициализации базовых компонентов проекта. Создание отдельных шаблонов для пересечения множеств технологий, механик и иных условий, как представлено в аналогах, может быть неэффективным. Во-первых, потребуется достаточное время на создание каждой из таких заготовок. Во-вторых, не все варианты могут быть востребованы командой. В-третьих, выбор шаблонов из имеющегося множества также станет непростой задачей (так сделано в системе Asana). Для задач должна быть доступна настройка всех параметров, включая перечень меток, что позволяет к моменту начала работ иметь начальное представление проекта с назначенными исполнителями и установленным порядком решения зависимых задач. Для достижения большего эффекта в режиме администратора должны быть доступны настройка опросных листов и создание отдельных опросных листов (например, для приложений виртуальной реальности). Тогда решение становится

универсальным для всей сферы разработки видеоигр.

Обсуждение реализации

Для проверки разработанной методики новые функции (генерации шаблона проекта с задачами и заметками по результатам заполнения брифа, автоматического сбора и передачи результатов работы другим участникам команды разработки, автоматизации дублирования задач и назначения исполнителей с учетом процесса разработки проектов) были внедрены в систему LeanTime [21] с открытым исходным кодом, написанным на языке PHP. Данная система опережает другие открытые системы по реализованному функционалу и отвечает почти всем требованиям современного таск-трекера (возможность ведения разработки как по Waterfall, так и по Agile; наличие канбандоски для отслеживания перечня задач; возможность интеграции с системами хранения больших объемов данных – для хранения результатов по итогам выполнения задачи), для полного соответствия этому термину были внедрены заметки, зависимость задач и диаграмма Ганта по задачам (рис. 4).

Более того, была подтверждена гипотеза упрощения таск-трекинга на примере ведения проекта в реализованной системе и ее аналоге (Asana). Согласно собранной статистике, предложенный подход к автоматизации показал сокращение времени на управленческие задачи более чем на 20%. Однако в случае потребности множества новых шаблонов сокращение может превысить и 40%. Действительно, когда создается один проект либо используется только один шаблон, время, затрачиваемое на создание системы проектных ролей и меток, а также составление опросного листа, бессмысленны. Ручные настройки исполнителей задач и зависимостей между задачами также оказывают существенное влияние на сокращение времени менеджера.

Возможность передачи артефактов разработки также может быть необязательным требованием к системе управления задачами, однако такой функционал позволит оптимизировать взаимодействие сотрудников внутри команды.

Стоит отметить, что немаловажным фактором стало наличие большого объема релевантного опыта у авторов статьи [22–24].

Разрабатываемый инструмент можно соединить с другими инструментами автоматизации, которые встраиваются в пайплайн гейм-

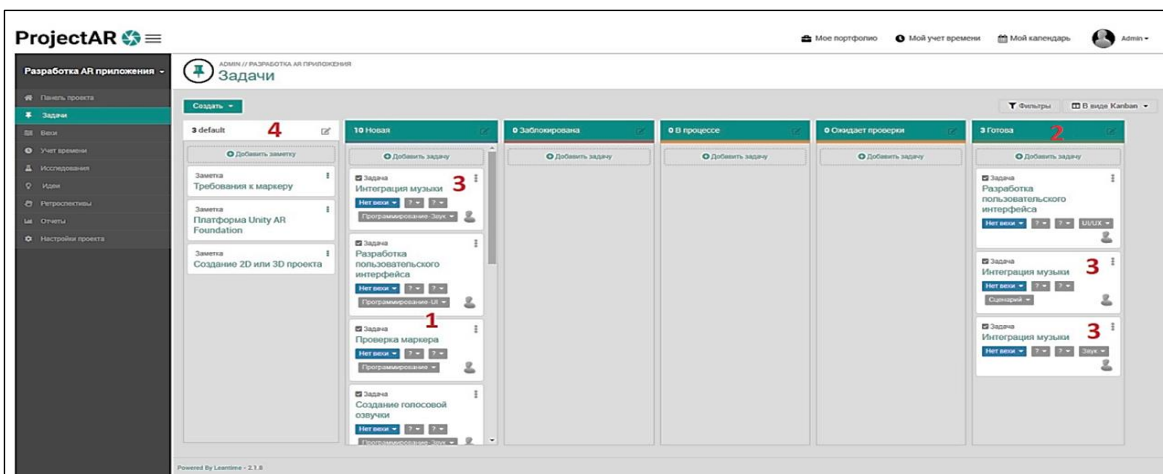


Рис. 4. Интерфейс расширенного таск-трекера: 1 – отображение меток задач; 2 – при переносе в статус «Готово» зависящим задачам назначается исполнитель по роли для метки; 3 – сгенерированные дубликаты одной задачи для уникальных меток с учетом последовательности выполнения; 4 – заметки к проекту

Fig. 4. The extended task tracker interface: 1 – displaying task labels; 2 – dependent tasks are assigned an executor according to the role for the label when moving to the “Done” status; 3 – generated duplicates of one task for unique labels taking into account the execution sequence; 4 – project notes

дев-проектов. Например, интеграция артефактов генератора игровых прототипов [25, 26] может производиться на нескольких уровнях: сценарий, компьютерная графика, код.

Заключение

Несмотря на наличие множества систем таск-трекинга, как для сферы разработки игр, так и для традиционной разработки ПО существуют пути дальнейшего расширения функционала. Создание собственных подходов к уже реализованным возможностям, которые учитывают специфику и процессы разработки проекта, позволит сократить время менеджеров на управление задачами, а также такие риски, как

проблемы коммуникации внутри команды.

В статье предложен новый подход к ведению проекта в системах таск-трекинга, учитывающий особенности разработки геймдев-проектов и включающий в себя генерацию проекта по его ключевым вопросам, автоматизацию назначения исполнителей с учетом их роли и последовательности выполнения задач, а также передачу артефактов разработки по цепочке последовательных задач.

Необходимо отметить, что созданный веб-инструмент можно использовать в качестве инструмента автоматизации процессов разработки в любых узкоспециализированных ИТ-направлениях, не ограничиваясь лишь управлением геймдев-проектами.

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

Литература

1. Vasudevamurt V., Uskov A. Serious game engines: Analysis and applications. Proc. IEEE Int. Conf. EIT, 2015, pp. 440–445. DOI: 10.1109/EIT.2015.7293381.
2. Aleem S., Capretz L., Ahmed F. Game development software engineering process life cycle: A systematic review. J. of Software Engineering Research and Development, 2016, vol. 4, art. 6. DOI: 10.1186/s40411-016-0032-7.
3. Murphy-Hill E., Zimmermann T., Nagappan N. Cowboys, ankle sprains, and keepers of quality: How is video game development different from software development? Proc. ICSE, 2014, pp. 1–11. DOI: 10.1145/2568225.2568226.
4. Oliveira J., Tereso A., Machado R. An application to select collaborative project management software tools. In: New Perspectives in Information Systems and Technologies, vol. 1. AISC, 2014, vol. 275, pp. 467–476. DOI: 10.1007/978-3-319-05951-8_44.

5. Andonova S., Bontchev B. Web project tracker. Proc. Int. Conf. CompSysTech, 2003, pp. 208–213. DOI: 10.1145/973620.973655.
6. Abramova V., Pires F., Bernardino J. Open source vs proprietary project management tools. In: New Advances in Information Systems and Technologies. AISC, 2016, pp. 331–340. DOI: 10.1007/978-3-319-31232-3_31.
7. Ferreira T., Gutiérrez-Artacho J., Bernardino J. Freemium project management tools: Asana, freedcamp and ace project. In: Trends and Advances in Information Systems and Technologies. AISC, 2018, pp. 1026–1037. DOI: 10.1007/978-3-319-77703-0_100.
8. El-Migid M.-A.A., Cai D., Niven T., Vo J., Madampe K., Grundy J., Hoda R. Emotimonitor: A Trello power-up to capture and monitor emotions of Agile teams. J. of Systems and Software, 2022, vol. 186, art. 111206. DOI: 10.1016/j.jss.2021.111206.
9. Lyulina E., Birillo A., Kovalenko V., Bryksin T. TaskTracker-tool: A toolkit for tracking of code snapshots and activity data during solution of programming tasks. Proc. SIGCSE '21, 2021, pp. 495–501. DOI: 10.1145/3408877.3432534.
10. Hall M. Choosing a Project Management Tool for Game Development. 2018. URL: <https://www.gamedeveloper.com/business/choosing-a-project-management-tool-for-game-development> (дата обращения: 19.07.2022).
11. Monday.com. Build Your Own Custom Automation. URL: <https://support.monday.com/hc/en-us/articles/360012254440-Build-your-own-custom-automation> (дата обращения: 19.07.2022).
12. Trello. Автоматизация любых действий в Trello. URL: <https://trello.com/ru/guide/automate-anything> (дата обращения: 19.07.2022).
13. ATlassian. Jira Software. Руководство 3. Расширение Jira Software. Автоматизация: примеры использования. URL: <https://www.atlassian.com/ru/software/jira/guides/expand-jira/automation-use-cases/> (дата обращения: 19.07.2022).
14. Asana Guide. Исполнители и участники задач. URL: <https://asana.com/ru/guide/help/tasks/people> (дата обращения: 19.07.2022).
15. Asana Guide. Создание и использование шаблонов Asana. URL: <https://asana.com/ru/guide/team/advanced/create-use-asana-templates> (дата обращения: 19.07.2022).
16. HacknPlan. Setting up a Project. Project Information. URL: https://hacknplan.com/knowledge-base/setting-up-a-project/#articleTOC_2 (дата обращения: 19.07.2022).
17. PHYGITALISM. AR механики: все о дополненной реальности. 2021. URL: <https://phygitalism.com/wp-content/uploads/2021/04/PHYGITALISM-White-Paper-AR-2021.pdf> (дата обращения: 19.07.2022).
18. Godoy A., Barbosa E. Game-Scrum: An approach to agile game development. Proc. of SBGames, 2010, pp. 292–295.
19. Petrillo F., Pimenta M., Trindade F., Dietrich C. Houston, we have a problem ...: A survey of actual problems in computer games development. Proc. ACM SAC, 2008, pp. 707–711. DOI: 10.1145/1363686.1363854.
20. Callele D., Neufeld E., Schneider K. Requirements engineering and the creative process in the video game industry. Proc. XIII IEEE Int. Conf. RE, 2005, pp. 240–250. DOI: 10.1109/RE.2005.58.
21. Sofbox. Innovation Management for Remote Teams. Open Source Project Management System – Leantime. URL: <https://leantime.io/> (дата обращения: 19.07.2022).
22. Бакиров А.Р., Кугуракова В.В., Манахов Н.Р., Селезнёва Н.Э. Использование контроллера Microsoft Kinect в разработке реабилитационных игр // Электронные библиотеки. 2016. Т. 19. № 6. С. 521–537.
23. Кугуракова В.В., Судаков В.А., Нургалиев Д.К., Шараева Р.А. и др. Программа для симуляции трубы горения в виртуальной реальности: Свид. о регистр. ПрЭВМ № 2020660114. Рос. Федерация, 2020.
24. Sultanova R., Sharaeva R. Virtual reality-based immersive simulation mechanics for invasive surgery training. Proc. Int. Conf. DeSE, 2019, pp. 924–928. DOI: 10.1109/DeSE.2019.00171.
25. Сахибгареева Г.Ф., Кугуракова В.В. Прототипирование вариативности сюжета компьютерных игр // Научный сервис в сети Интернет: тр. XXIII Всеросс. науч. конф. 2021. С. 347–360. DOI: 10.20948/abrau-2021-11.
26. Sahibgareeva G., Kugurakova V. Branched Structure component for a video game scenario prototype generator. Proc. XXIII Sci. Conf. Scientific Services & Internet, 2021, pp. 101–111. DOI: 10.20948/abrau-2021-10-ceur.

A method for simplifying task tracking in video game development projects

R.A. Sharaeva¹, Junior Researcher, r.sharaeva3496@gmail.com

V.V. Kugurakova¹, Ph.D. (Engineering), Associate Professor, Head of Laboratory,
vlada.kugurakova@gmail.com

N.E. Selezneva¹, Postgraduate Student, nataliseleznewa306@gmail.com

¹ Kazan Federal University, Institute of Information Technology and Intelligent Systems,
Department of Software Engineering, Kazan, 420008, Russian Federation

Abstract. Computer game development differs from “traditional” IT project development mainly due to the high degree of process creativity, which should be taken into account when applying software engineering methods.

The research topic is studying the specifics of project management process automation and developing a tool embodying the methods and recommendations formed for the game industry. The authors have analyzed topical scientific sources to set the goal and objectives correctly. As a result, they have highlighted a number of functions necessary for team task trackers that are: assigning a role or a task to an employee, creating dependent tasks, Gantt chart, giving access to information about related tasks to an employee, distributing links and documents, creating reusable templates.

Based on the extensive research results, the paper presents a methodology that simplifies the work of the project manager in terms of task management. The approach includes creation of the following features: executors automatically receive tasks according to their field of concern; automatic creation of duplicates and configuration of task dependencies; automatic transfer of artifacts to dependent tasks; generating a project template according to a brief. The work is interesting for the field of interactive project development with a prevailing game-dev component.

In order to check the result effectiveness, the generated practical approaches were integrated into the LeanTime system. The proposed approach has shown the reduction of time for management tasks by more than 20%; in case of need for many new templates the reduction can even exceed 40%. The designed web-tool can then be integrated as an automation tool not only in the development processes of gamedev projects, but in general for any highly specialized IT development areas.

Keywords: software engineering, IT development, gamedev, task tracker, automation.

Acknowledgements. This paper has been supported by the Kazan Federal University Strategic Academic Leadership Program (“PRIORITY-2030”).

References

1. Vasudevamurt V., Uskov A. Serious game engines: Analysis and applications. *Proc. IEEE Int. Conf. EIT*, 2015, pp. 440–445. DOI: 10.1109/EIT.2015.7293381.
2. Aleem S., Capretz L., Ahmed F. Game development software engineering process life cycle: A systematic review. *J. of Software Engineering Research and Development*, 2016, vol. 4, art. 6. DOI: 10.1186/s40411-016-0032-7.
3. Murphy-Hill E., Zimmermann T., Nagappan N. Cowboys, ankle sprains, and keepers of quality: How is video game development different from software development? *Proc. ICSE*, 2014, pp. 1–11. DOI: 10.1145/2568225.2568226.
4. Oliveira J., Tereso A., Machado R. An application to select collaborative project management software tools. In: *New Perspectives in Information Systems and Technologies*, vol. 1. AISC, 2014, vol. 275, pp. 467–476. DOI: 10.1007/978-3-319-05951-8_44.
5. Andonova S., Bontchev B. Web project tracker. *Proc. Int. Conf. CompSysTech*, 2003, pp. 208–213. DOI: 10.1145/973620.973655.
6. Abramova V., Pires F., Bernardino J. Open source vs proprietary project management tools. In: *New Advances in Information Systems and Technologies*. AISC, 2016, pp. 331–340. DOI: 10.1007/978-3-319-31232-3_31.
7. Ferreira T., Gutiérrez-Artacho J., Bernardino J. Freemium project management tools: Asana, freedcamp and ace project. In: *Trends and Advances in Information Systems and Technologies*. AISC, 2018, pp. 1026–1037. DOI: 10.1007/978-3-319-77703-0_100.

8. El-Migid M.-A.A., Cai D., Niven T., Vo J., Madampe K., Grundy J., Hoda R. Emotimonitor: A Trello power-up to capture and monitor emotions of Agile teams. *J. of Systems and Software*, 2022, vol. 186, art. 111206. DOI: 10.1016/j.jss.2021.111206.
9. Lyulina E., Birillo A., Kovalenko V., Bryksin T. TaskTracker-tool: A toolkit for tracking of code snapshots and activity data during solution of programming tasks. *Proc. SIGCSE '21*, 2021, pp. 495–501. DOI: 10.1145/3408877.3432534.
10. Hall M. *Choosing a Project Management Tool for Game Development*. 2018. Available at: <https://www.gamedeveloper.com/business/choosing-a-project-management-tool-for-game-development> (accessed July 19, 2022).
11. *Monday.com. Build Your Own Custom Automation*. Available at: <https://support.monday.com/hc/en-us/articles/360012254440-Build-your-own-custom-automation> (accessed July 19, 2022).
12. *Trello. Automation of Any Actions in Trello*. Available at: <https://trello.com/ru/guide/automate-anything> (accessed July 19, 2022).
13. *ATLASSIAN. Jira Software. Guide 3. Jira Software Extension. Automation: Common Use Cases*. Available at: <https://www.atlassian.com/ru/software/jira/guides/expand-jira/automation-use-cases/> (accessed July 19, 2022).
14. *Asana Guide. Task Assignee & Collaborators*. Available at: <https://asana.com/ru/guide/help/tasks/people> (accessed July 19, 2022).
15. *Asana Guide. Creating and Using Asana Templates*. Available at: <https://asana.com/ru/guide/team/advanced/create-use-asana-templates> (accessed July 19, 2022).
16. *HacknPlan. Setting up a Project. Project Information*. Available at: https://hacknplan.com/knowledge-base/setting-up-a-project/#articleTOC_2 (accessed July 19, 2022).
17. *PHYGITALISM. AR Mechanics: Everything about Augmented Reality*. Available at: <https://phygitalism.com/wp-content/uploads/2021/04/PHYGITALISM-White-Paper-AR-2021.pdf> (accessed July 19, 2022).
18. Godoy A., Barbosa E. Game-Scrum: An approach to agile game development. *Proc. of SBGames*, 2010, pp. 292–295.
19. Petrillo F., Pimenta M., Trindade F., Dietrich C. Houston, we have a problem ...: A survey of actual problems in computer games development. *Proc. ACM SAC*, 2008, pp. 707–711. DOI: 10.1145/1363686.1363854.
20. Calle D., Neufeld E., Schneider K. Requirements engineering and the creative process in the video game industry. *Proc. XIII IEEE Int. Conf. RE*, 2005, pp. 240–250. DOI: 10.1109/RE.2005.58.
21. *Sofbox. Innovation Management for Remote Teams. Open Source Project Management System – Leantime*. Available at: <https://leantime.io/> (accessed July 19, 2022).
22. Bakirov A.R., Kugurakova V.V., Manahov N.R., Seleznjova N.E. Using Microsoft kinect in the rehabilitation games development. *Russ. Digital Libraries J.*, 2016, vol. 19, no. 6, pp. 521–537 (in Russ.).
23. Kugurakova V.V., Sudakov V.A., Nurgaliev D.K., Sharaeva R.A. et al. *Software for Simulation of Combustion Pipe in Virtual Reality*. Patent RF, no. 2020660114, 2020 (in Russ.).
24. Sultanova R., Sharaeva R. Virtual reality-based immersive simulation mechanics for invasive surgery training. *Proc. Int. Conf. DeSE*, 2019, pp. 924–928. DOI: 10.1109/DeSE.2019.00171.
25. Sakhigareeva G.F., Kugurakova V.V. Prototyping the variability of computer games plots. *Proc. XXIII Sci. Conf. Scientific Services & Internet*, 2021, pp. 347–360. DOI: 10.20948/abrau-2021-11 (in Russ.).
26. Sahibgareeva G., Kugurakova V. Branched Structure component for a video game scenario prototype generator. *Proc. XXIII Sci. Conf. Scientific Services & Internet*, 2021, pp. 101–111. DOI: 10.20948/abrau-2021-10-ceur.

Для цитирования

Шараева Р.А., Кугуракова В.В., Селезнева Н.Э. Методика упрощения таск-трекинга в проектах игровой индустрии // Программные продукты и системы. 2022. Т. 35. № 3. С. 374–383. DOI: 10.15827/0236-235X.139.374-383.

For citation

Sharaeva R.A., Kugurakova V.V., Selezneva N.E. A method for simplifying task tracking in video game development projects. *Software & Systems*, 2022, vol. 35, no. 3, pp. 374–383 (in Russ.). DOI: 10.15827/0236-235X.139.374-383.