

Формализация требования для разработки ПО, решающего задачу автоматизации управления аналитикой трудозатрат сотрудников в современных системах управления проектами путем сравнительного анализа существующих подходов

Ключевые слова: управление ресурсами, менеджер задач, система управления проектами, трудозатраты, аналитика трудозатрат

Аннотация

В данной работе были рассмотрены различные инструменты, позволяющие обеспечить автоматизацию аналитики трудозатрат, и проведен сравнительный анализ систем управления проектами на наличие данных инструментов. По результатам анализа были выявлены недостатки выбранных систем, заключающиеся в отсутствии необходимых для проведения аналитики функций, которые были учтены при формировании требований для реализации предлагаемого веб-сервиса.

Введение

Управление ресурсами – одна из важнейших управленческих задач на предприятии любой отрасли. Эти ресурсы можно распределить на две основные группы: материальные и трудовые. Материальные ресурсы, такие как сырье, запасы, оборудование и другие, расходуются путем списания их количества или части стоимости на производство. Такие ресурсы измеримы, контролируются и управляемы.

Трудовые ресурсы, в свою очередь, представляют собой сотрудников предприятия, выполняющих определенную работу. Эти ресурсы, в отличие от материальных, расходуют рабочее время на выполнение работы. А измерение и контроль расходов рабочего времени – более сложная задача.

Для современных IT-компаний, занимающихся разработкой программных продуктов, трудовые ресурсы являются основными, поэтому важно уметь ими правильно распоряжаться.

В настоящее время большинство IT-компаний используют менеджеры задач для обеспечения управления процессами разработки и сопровождения своих продуктов. Данные программные системы, как правило, содержат разные инструменты для настройки потока задач и соответствующих бизнес-процессов, однако возможности по отслеживанию рабочей нагрузки сотрудников часто не реализованы на необходимом уровне. Ошибки в планировании распределения задач приводит к выгоранию сотрудников и несвоевременному выполнению обязательств компании, что может стать причиной финансовых потерь. Таким образом, важно использовать программное обеспечение, корректно оценивающее трудозатраты сотрудников и помогающее верно распределить ресурсы компании.

Для решения проблемы обеспечения автоматизации аналитики трудозатрат сотрудников с использованием современных систем управления проектами требуется рассмотреть существующие программные решения и формализовать требования для разработки соответствующего ПО. Объектом данного исследования являются системы управления

проектами. Предмет исследования - функциональность систем управления проектами по автоматизации аналитики трудозатрат сотрудников.

Обзор предметной области

Для проведения сравнительного анализа были выбраны часто используемые в индустрии [1] системы управления проектами:

Atlassian Jira

JIRA [2] – коммерческая система отслеживания ошибок и управления проектами с закрытым исходным кодом, разрабатываемая компанией Atlassian. Разработана преимущественно на языке программирования Java и поддерживает интеграции со сторонними сервисами с помощью REST API или системных расширений. Распространяется как для самостоятельной установки пользователем, так и как SaaS (Software as a Service)

Easy Redmine

Easy Redmine [3] – коммерческая версия системы управления проектами с открытым исходным кодом Redmine с дополненными функциональными возможностями, разрабатываемая одноименной компанией. Easy Redmine является программным обеспечением с закрытым исходным кодом. Данный программный продукт создан при помощи языка программирования Ruby и веб-фреймворка Ruby on Rails, а также поддерживает сторонние интеграции с помощью REST API. Распространяется как для самостоятельной установки пользователем, так и как SaaS (Software as a Service)

Trello

Trello [4] – коммерческая система управления проектами с закрытым исходным кодом, разрабатываемая компанией Atlassian. В данной системе используется язык программирования JavaScript и веб-фреймворк React, а распространение происходит только по модели SaaS (Software as a Service). Поддерживаются сторонние решения и возможность работы с данными средствами REST API

JetBrains YouTrack

YouTrack [5] – коммерческая система отслеживания ошибок и управления проектами, разрабатываемая компанией JetBrains. Разработка системы ведется на языке программирования Java и поддерживает интеграции со сторонними сервисами с помощью REST API. Распространяется как для самостоятельной установки пользователем, так и как SaaS (Software as a Service)

Azure Boards

Azure Boards [6] является частью системы Azure DevOps Server (ранее Microsoft Team Foundation Server) – продукта компании Microsoft, представляющую собой комплексное решение для совместной работы над проектами по разработке программного обеспечения. Присутствует поддержка ряда системных расширений, однако добавление сторонних интеграций не предусмотрено

Asana

Asana [7] – коммерческая система управления проектами, разрабатываемая компанией Asana. Система имеет закрытый исходный код, однако предоставляет API для разработчиков. Разработка ведется с использованием обширного набора технологий, среди которых TypeScript, React и другие

Для сравнения вышеуказанных систем были выбраны следующие функциональные возможности систем управления проектами:

Цена использования за одного пользователя

При выборе сервиса необходимо учитывать стоимость его использования компанией, чтобы планировать использование бюджета. Общая стоимость рассчитывается на основе цены за одного сотрудника, умноженного на число пользователей системы

Установка оценочных и фактических трудозатрат по задаче

Аналитика трудозатрат производится на основе данных по использованию сотрудниками трудовых ресурсов компании, поэтому важно иметь возможность установить первоначальную оценку и фактические трудозатраты сотрудника по задаче для последующей аналитики

Просмотр таблицы нормирования труда (разницы оценочного и фактического времени выполнения задач за период)

Таблица нормирования труда позволяет определить сотрудников, превысивших оценочные трудозатраты по задачам для последующего выяснения причин отклонений и их устранения. Кроме того, эти данные позволяют рассчитать реальную трудоемкость различных операций

Просмотр статистики “сгорания” времени выполнения проекта (“Burndown chart”)

Диаграмма сгорания задач (Burndown Chart) используется в гибких методологиях разработки, таких как Agile и Scrum, и позволяет оценить степень выполнения задач командой по отношению к прошедшему времени, количество закрытых и нерешенных задач за единицу времени

Таблица 1. Сравнение аналогов

Система управления проектами	Jira	Easy Redmine	Trello	YouTrack	Azure Boards	Asana
Цена использования за одного пользователя	7\$	3\$	9\$	1\$	6\$	11\$
Установка оценочных и фактических трудозатрат по задаче	Да	Да	Нет*	Да	Да	Нет*
Просмотр таблицы нормирования труда	Нет*	Да	Нет	Нет*	Нет	Нет
Просмотр статистики “сгорания” времени выполнения проекта	Да	Нет*	Нет	Да	Да	Нет

* Да, при использовании сторонних расширений

По итогам проведенного сравнительного анализа можно сделать вывод о том, что рассмотренные системы не полностью реализуют указанные в критериях сравнения функциональные возможности, необходимые для проведения аналитики трудозатрат сотрудников. Кроме того, стоимость использования сервисов не всегда оправдана, т.к. прямой зависимости между ценой использования за одного пользователя и количеством функций не наблюдается

Выбор метода решения

Для решения задачи автоматизации управления аналитикой трудозатрат сотрудников предлагается реализовать веб-портал, имеющий интеграции с вышеуказанными системами управления проектами и реализующий следующие функции:

- Установка оценочных и фактических трудозатрат по задаче
- Просмотр таблицы нормирования труда
- Просмотр статистики “сгорания” времени выполнения проекта

Кроме того, разрабатываемый продукт должен распространяться по модели Open Source, тем самым обеспечив минимальную стоимость использования за одного сотрудника

Описание метода решения

Реализовать вышеуказанные функции предлагается с использованием следующих технологий:

1. Для хранения данных в системе (таких как данные о трудозатратах, информация о пользователях и другое) и управления ими предлагается использовать СУБД MongoDB версии не ниже 4.2 и обеспечить интеграцию с серверной частью приложения, реализованного с помощью JavaScript-фреймворка Express версии не ниже 4.17.1 и программной платформы NodeJS версии не ниже 13.0.0, с помощью Mongoose ODM версии не ниже 5.7.0.
2. Для упрощения процесса построения требуемых таблиц, а также для отображения и редактирования в них соответствующих данных предлагается использовать библиотеку Plotly Pivottable, позволяющая создавать сводные таблицы для отображения большого количества информации. Данная библиотека разработана для Javascript-фреймворка React версии не ниже 16.0, который предлагается использовать для построения клиентской части приложения. Данная библиотека позволит реализовать критерии “Установка оценочных и фактических трудозатрат по задаче” и “Просмотр таблицы нормирования труда” и обеспечить наличие соответствующих таблиц.
3. Для построения различных графиков в системе (таких как график “сгорания” времени выполнения проекта) предлагается использовать Javascript-библиотеку Chart.JS версии не ниже 2.9.0., позволяющая строить гистограммы и линейные графики. Данная библиотека позволит реализовать критерий “Посмотр статистики”сгорания" времени выполнения проекта" и обеспечить наличие соответствующего графика.
4. Для обмена данными между клиентской и серверной частью приложения предлагается использовать стандарт GraphQL, реализованный в библиотеках Apollo Client (версии не ниже 3.1.0) и Apollo Server (версии не ниже 2.8.1) для соответствующих частей приложения. Использование технологии GraphQL вместо REST позволит сэкономить время на реализацию API серверного приложения и облегчить настройку взаимодействия между подсистемами. Кроме того, использование MongoDB в качестве базы данных позволит упростить интеграцию GraphQL, так как оба стандарта рассчитаны на работу с иерархическими структурами данных (документные хранилища)

Таким образом, вышеуказанный набор технологий позволит реализовать поставленную задачу за счет реализации необходимых критериев. Кроме того, выбор JavaScript в качестве языка разработки как для клиентской, так и для серверной части приложения, позволит упростить процесс разработки за счет использования общей кодовой базы. А использование контейнеризации с помощью Docker позволит упростить распространение приложения в Open Source сообществе, тем самым повышая его конкурентоспособность

Заключение

По результатам проведенного сравнительного анализа были определены слабые места в функциональности ряда менеджеров задач по автоматизации аналитики управления трудозатратами. Акцентируя внимание на необходимости правильного управления трудовыми ресурсами, был выдвинут ряд критериев, сформировав тем самым требования для разработки веб-сервиса. Кроме того, был предложен набор технологий для разработки веб-сервиса, решающего поставленную задачу. Данный набор был собран таким образом, что его

использование обеспечивает гибкость и масштабируемость разрабатываемого программного продукта.

В будущем планируется реализация веб-сервиса и его апробация на реальных проектах.

Список литературы

1. Alla Kravets. “Web Portal for Project Management in Electronics Design Software Development” 2019 International Seminar on Electron Devices Design and Production (SED). IEEE, 2019.
2. Jira Software Features. Доступно на: <https://www.atlassian.com/software/jira/features> (посещено 29 ноября 2019)
3. Easy Redmine 2019 Features. Доступно на: <https://www.easyredmine.com/software/easy-redmine> (посещено 29 ноября 2019)
4. Trello Features (Trello Board). Доступно на: <https://trello.com/b/zo2N0vE6/trello-features> (посещено 29 ноября 2019)
5. Features. YouTrack - JetBrains. Доступно на: <https://www.jetbrains.com/youtrack/features/> (посещено 29 ноября 2019)
6. Azure Boards documentation. Доступно на: <https://docs.microsoft.com/en-us/azure/devops/boards/?view=azure-devops> (посещено 29 ноября 2019)
7. Full list of Asana’s features. Доступно на: <https://asana.com/guide/get-started/share/features> (посещено 29 ноября 2019)