

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники
Кафедра инженерной психологии и эргономики

УДК 004.4'4

Федюнькин
Владислав Васильевич

ПРОГРАММНАЯ ПЛАТФОРМА ДЛЯ СОЗДАНИЯ СОБСТВЕННЫХ
ПЛАГИНОВ И РАСШИРЕНИЙ ATlassian JIRA

АВТОРЕФЕРАТ

на соискание академической степени
магистра технических наук

1-23 80 08 – Психология труда, инженерная психология, эргономика

Магистрант В.В. Федюнькин
Группа 715023

Научный руководитель
И.Т. Кавецкий, кандидат
психологических наук, доцент

Заведующий кафедрой ИПиЭ
К.Д. Яшин, кандидат технических
наук, доцент

Нормоконтролер
В.С. Гладкая, магистр
технических наук
ассистент кафедры ИПиЭ

Минск 2018

ВВЕДЕНИЕ

Процесс разработки программного обеспечения постоянно претерпевает значительные изменения. Основной причиной является увеличение сложности самих проектов и требуемой инфраструктуры для их реализации. Команды разработки становятся больше, а в общую структуру включаются новые роли.

С ростом сложности реализуемых проектов и увеличением количества доступных технологий разработки появилась необходимость в бизнес анализе и контроле качества. Эти этапы помогали удостовериться, что полученный продукт соответствовал ожиданиям заказчика и не содержал ошибок в функциональной части. Основным минусом было то, что проверка проводилась только в начале и в конце производственного процесса.

Описанная модель хорошо справлялась с маленькими продуктами, однако вызывала много проблем на крупных проектах, так как заказчик мог изъявить желание внести какие-либо правки или изменения, чего нельзя было сделать из-за зафиксированного технического задания. Решением данной проблемы послужил переход от привычной потоковой практики к гибкой методологии управления проектами. Гибкая методология совершила концептуальный переворот в понимании правильного подхода к разработке программного обеспечения. Стали появляться частные реализации идеи гибкого подхода.

Введение гибкой методологии позволило упростить взаимодействие между заказчиком и исполнителями, вносить изменения в программный продукт в процессе разработки, проводить контроль качества на промежуточных этапах разработки.

Описанная технология требовала квалифицированных сотрудников и удобных инструментов менеджмента процессов. В результате возник целый ряд систем управления проектами. Они выполняли одну задачу, но у каждого менеджера было свое видение правильного управления проектами, в результате чего не получалось разработать универсальный, подходящий всем продукт. Решением стала возможность пользовательской настройки конечного продукта под свои нужды с использованием плагинов и дополнений.

Таким образом было решено сделать максимально детализированную систему управления проектами с большим количеством подключаемых модулей и плагинов, набор которых мог быть расширен своими усилиями. Актуальность данной работы обусловлена общими потребностями рынка разработки программного обеспечения, растущей популярностью философии гибких подходов к разработке и распространенностью Jira как одной из основных систем управления проектами.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Целью магистерской диссертации является разработка программной платформы, которая позволит создавать собственные плагины и приложения для Jira, а также использование полученной платформы для создания нескольких пробных плагинов.

Объектом исследования является система управления проектами – Jira. Предметом исследования является возможность расширения и дополнения базового функционала указанной системы.

В разработанной программной платформе представлен широкий спектр вспомогательного функционала для пользователей, включающий в себя интеграцию с внешними сервисами Jira, прямое соединение с её базой данных, интеграцию с дополнительными средствами для сложных математических расчетов и построения графиков. Благодаря полученным возможностям значительно сокращаются временные затраты разработчиков дополнений и плагинов, а также появляется возможность настройки системы Jira под любой проект.

Гибкая настройка системы Jira, в свою очередь, упрощает взаимодействие компаний и менеджеров проектов с заказчиками, делая процесс разработки более прозрачным и понятным для широкого круга лиц.

Разработанная программная платформа была опробована для создания двух плагинов, позволяющих создавать расширенные диаграммы выгорания работ и производительности, и показала свою эффективность и практичность.

Диссертационная работа состоит из введения, перечня условных обозначений и терминов, общей характеристики работы, трех глав, заключения, списка использованной литературы и двух приложений. Общий объем работы составляет 58 страниц. Из них 35 страниц текста, 21 иллюстрация на 9 страницах, 2 таблиц на 2 страницах, список использованной литературы на 5 страницах, 2 приложения на 7 страницах. Было использовано 57 библиографических источников.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

В общей характеристике работы и введении определены цели и задачи работы, обоснована её актуальность и описан процесс становления гибкой методологии как нового стандарта разработки программного обеспечения. Также определены основные преимущества гибкой методологии над старым принципом работы по техническим заданиям.

В первой главе описаны основные положения гибкой методологии, представлены часто используемые термины и составы задействованных команд, включая распределение ролей и обязанности участников. Отдельно описан жизненный цикл разработки программного обеспечения в гибкой методологии, способы оценки задач и распределение задач по времени и между участниками команд. Подробно рассмотрены частные реализации гибкой методологии – скрам и канбан, определены различия и преимущества каждого из них, а также обоснована необходимость использования систем управления проектами при работе с ними. Дополнительно обусловлена необходимость использования смешанных методологий в особо крупных проектах.

Проведен сравнительный анализ наиболее популярных систем управления проектами – Trello, TeamWork, Atlassian Jira. Представлены примеры и описания пользовательских интерфейсов для каждой системы, выделены преимущества и недостатки, определены области возможного использования и способности к интеграции со сторонними ресурсами – базами данных, мессенджерами, репозиториями и хранилищами информации. Подробно рассмотрена система Atlassian Jira, проанализированы её функциональные возможности, отдельно выделены частота обновлений, широкая пользовательская база и возможность персональной настройки и расширения при помощи плагинов и дополнений. Также проанализирована возможность коммерческого распространения собственных расширений через встроенный магазин Atlassian с учетом выгодного разделения доходов, 25 к 75 в пользу разработчиков, а также доступной и простой системы лицензирования.

Во второй главе проанализированы базовые возможности Jira по отслеживанию прогресса отдельных задач и проекта в целом, обосновывается необходимость введения новых графиков – расширенной диаграммы выгорания работ и расширенной диаграммы производительности. Предоставлены необходимые математические формулы, которые будут использованы при расчетах, а также примеры полученных решений. Подробно описаны возможности новых диаграмм, их назначение и способы применения на проектах.

Определены основные функции, которые должны быть реализованы для эффективного взаимодействия пользователя с системой:

- генерация страниц с набором полей для ввода пользовательских данных;
- ввод входных данных согласно требованиям используемого плагина;
- сбор пользовательских данных, необходимых для функционирования плагина, и передача их в систему;
- произведение расчетов согласно полученным данным и используемому плагину;
- отображение результатов в виде цифровых значений в таблицах;
- отображение результатов в виде графиков и диаграмм;
- автоматический подбор необходимых значений для масштабирования полученных графиков и диаграмм;
- изменение входных фильтров после получения результатов;
- перерасчет данных в случае изменения пользовательского ввода;
- автоматический перерасчет данных в случае изменения внутренней структуры данных, хранимых в системе;
- ограничение доступа отображаемых данных в зависимости от роли пользователя в системе;
- валидация ошибок, связанных с некорректным вводом пользовательских данных;
- обработка ошибок, связанных с расчетами внутри системы или с уровнем доступа пользователя;
- отслеживание и сохранение пользовательских действий в системе;
- сохранение полученных результатов в виде экспортируемых файлов.

Приводится подробное описание для каждой функции. Произведено распределение описанных функций между человеком, машиной и средой, определен алгоритм работы оператора в полученной системе.

Сформированы эргономические требования, которым должна соответствовать разрабатываемая система. На основе полученных требований сформирована функциональная схема планируемого решения, определены основные используемые архитектурные компоненты, предоставляемые Jira.

Дополнительно определены эргономические требования к разрабатываемым диаграммам и графикам, исследованы возможности их реализации и внедрения. Установлен ряд требований для программной части разрабатываемого решения, необходимых для дальнейшего поддержания и развития продукта

В третьей главе обосновывается выбор языка программирования Java, среды разработки IntelliJ IDEA, а также остальных инструментов, обеспечивающих желаемых функционал. Описывается способ добавления разработанных решений к системе Jira.

Перечислены требования Jira к программной структуре, необходимые для успешного подключения плагина или дополнения к системе. После чего

подробно описаны выбранные инструменты, их назначение и примеры применения. Основными критериями для выбора являлись наличие подробной документации, лицензионные условия использования и широкий перечень функционала. Отдельно была рассмотрена возможность прямого взаимодействия программного продукта с базой данных, определены программные и системные требования для обеспечения данного процесса. После анализа всех инструментов была сформирована конечная архитектурная схема разрабатываемого продукта.

В ходе разработки представлена структура полученного программного решения, описаны основные функциональные элементы и их назначение. Обусловлена необходимость использования мавен-репозитория. Представлены примеры создаваемых классов, их функции и возможности. Отдельно выделен класс, отвечающий за создание графиков и диаграмм, описаны используемые программные решения, упрощающие чтение и написание кода, приведен пример создания одного из графиков. Описаны этапы документации и упаковки программного комплекса в надлежащие архивы.

После создания основного программного комплекса – описываются шаги по разработке собственных плагинов, в частности создание первичного шаблона, обработка ошибок, логирование и генерация конечных диаграмм. После чего описываются этапы подключения плагинов к системе Jira и их тестирование.

ЗАКЛЮЧЕНИЕ

В данной магистерской диссертации был произведен сравнительный анализ самых популярных систем управления проектами, выделены основные преимущества Jira перед конкурентами. После чего был проведен анализ базового функционала предоставляемого системой, а также исследованы возможности и направления для её расширения с помощью плагинов и дополнений.

Были сформированы требования к разрабатываемому программному комплексу, представлены необходимые математические формулы, а также эргономические параметры. Используя полученные данные была разработана функциональная схема и программная архитектура проекта. Также была исследована возможность коммерческой реализации и распространения продукта с использованием встроенного магазина Atlassian.

После чего был разработан программный комплекс, позволяющий создавать плагины и дополнения для Jira любой сложности и конструкции. Полученный комплекс значительно уменьшает временные затраты разработчика на нахождение новых данных, их компоновку и визуальное отображение. Он не замедляет работу общей системы, предоставляет подробный обработчик ошибок и детальное логирование.

Программный комплекс разработан на языке Java с использованием среды разработки IntelliJ IDEA. Для реализации вспомогательного функционала были использованы сторонние бесплатные библиотеки, упрощающие и ускоряющие математические расчеты, отвечающие за генерирование графиков и диаграмм, а также предоставляющие доступ к интерфейсам Jira и базе данных. Для написания основной части кода использовались возможности языка Java, популярные шаблоны проектирования, а также средства Atlassian SDK. Полученное решение легко расширяемо и соответствует всем актуальным потребностям разработчиков, а также требованиям для размещения в магазине Atlassian.

Комплекс предоставляет пользователям следующие возможности:

- получение широкого спектра данных из нескольких интерфейсов, присутствующих в любой версии Jira;
- получение доступа к базе данных Jira, чего не было представлено в исходном API;
- построение различных графиков и диаграмм по переданным параметрам с использованием различных библиотек;
- добавление в систему Jira новых полей, переменных и прочих структурных единиц.

Для подтверждения работоспособности и эффективности полученного программного комплекса с его помощью были созданы расширенные диаграммы выгорания задач и производительности, проверена и подтверждена их точность, произведена интеграция с официальной коммерческой версией Jira.

Теоретическая часть, а также промежуточные результаты разработки были представлены на 54 научной конференции аспирантов, магистрантов и студентов БГУИР.

На данный момент решение активно используется на корпоративном уровне в одной из аутсорсинговых компаний Минска. В ближайшее время планируется выход на коммерческий рынок.

Список публикаций соискателя

[1] Федюнькин, В.В. Программная платформа для создания собственных плагинов и расширений Atlassian Jira / В.В. Федюнькин // Программная платформа для создания собственных плагинов и расширений Atlassian Jira: Тезисы докл. к конф. – Минск, 2018