

УДК 004.01

УПРАВЛЕНИЕ ТРЕБОВАНИЯМИ И ВЕРСИОНИРОВАНИЕ ПРИ СМЕШАННОЙ ПРОЕКТНОЙ И ПРОДУКТОВОЙ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Герш Д.Д. (Университет ИТМО, ФТМИ)

Научный руководитель – профессор, доктор экономических наук Цуканова О.А.
(Университет ИТМО, ФТМИ)

В рамках исследования были сформулированы требования и рекомендации по набору документации для поддержки полного цикла разработки программного обеспечения при смешанной проектной и продуктовой разработке, выбрана технология и реализован предлагаемый подход.

Введение. При разработке программного обеспечения для документирования, отслеживания изменений и приоритезации используются системы управления требованиями. Часто эти системы интегрированы с другим ПО, применяющимся при разработке: системами управления проектами, системами отслеживания задач и ошибок (баг-трекерами) и другими.

Требования обязательно версионятся, так, как и разработчикам, и инженерам по тестированию и поддержке, и аналитикам необходимо иметь доступ к требованиям для определённой версии программного продукта и/или приложения.

Большая часть систем управления требований ориентирована на определенный тип разработки: проектный или продуктовый. В первом случае требования для проекта разрабатываются под конкретный проект, во втором случае они применимы для продукта и/или группы проектов.

Особенность смешанного подхода заключается в том, что есть ядро системы, так называемый продукт, который развивается как самостоятельно, так и в рамках проектов, базирующихся на ядре. В рамках одного базового продукта ведется разработка нескольких проектов, изменения вносятся одновременно, как в стволочной ветви (мастер-ветви) программного кода, так и в релизных и функциональных ветвях. При этом часть функциональных изменений являются проектно-специфичными и не затрагивают ядро системы.

При смешанном подходе становится затруднительным поддержание актуальной документации по продукту, а также по каждому проекту в отдельности.

Целью работы является формирование рекомендаций по управлению и версионированию требований при смешанной проектной и продуктовой разработке программного обеспечения.

Основная часть. Функциональные требования к программному продукту должны давать ответы на следующие вопросы:

1. Какие функциональные возможности доступны в данной версии продукта?
2. Какие функциональные возможности доступны в данной версии проекта?
3. Что было сделано в определённом релизе продукта?
4. Что было сделано в определённом релизе проекта?

Данный набор документации можно получить, соблюдая следующие рекомендации по написанию требований:

1. Для хранения требований должна использоваться система управления требованиями, поддерживающая версионирование
2. Должен использоваться репозиторий для хранения продуктовых требований, вносить изменения в который может аналитик любого проекта.
3. Каждый проект, реализуемый на базе продукта, должен иметь собственный репозиторий для хранения требований, уникальных для данного проекта. Вносить изменения в репозиторий может только аналитик данного проекта.

4. Каждый описанный требованиями вариант использования должен быть связан с реализующей его задачей в системе отслеживания задач и ошибок, чтобы однозначно можно было определить статус данного требования: новое, в разработке, в тестировании, реализовано.
5. Для каждого репозитория требований должны создаваться базовые версии – версия, фиксирующая срез требований для конкретного релиза.

В рамках доклада предлагается рассмотреть решение поставленной проблемы с использованием технологий компании Atlassian, которая разрабатывает JIRA – одну из самых функциональных и известных систем отслеживания задач и ошибок.

В качестве репозитория для хранения требований предлагается использовать систему Confluence, поддерживающую версионирование и интегрированную с системой JIRA. Для создания базовых версий требований использовано дополнение Requirements Yogi.

Выводы. Предлагаемое решение было внедрено в крупной компании-разработчике программного обеспечения. Внедренное решение позволило решить следующие задачи:

1. Инженеры по тестированию имеют полноценную документацию для проведения регрессионного тестирования, как продукта, так и проекта.
2. У всех участников проекта есть доступ к документации, описывающей состояние для всех актуальных версий системы:
 - a. Тестируемая версия.
 - b. Версия, проходящая приемо-сдаточные испытания на стороне заказчика.
 - c. Версия, доступная конечным пользователям.

К минусам данного подхода можно отнести увеличение времени на поддержание документации аналитиками.