

КЛЮЧЕВЫЕ ОСОБЕННОСТИ ПОДХОДА К РЕАЛИЗАЦИИ FINTECH ПРОДУКТОВ

Михальчук В.Д. (Факультет технологического менеджмента и инноваций,
Национальный исследовательский университет ИТМО,
г. Санкт-Петербург, Российская Федерация),

Научный руководитель – к.э.н., доцент Торосян Е.К.

(Факультет технологического менеджмента и инноваций, Национальный исследовательский
университет ИТМО,
г. Санкт-Петербург, Российская Федерация)

Аннотация. Современные команды по разработке ПО придерживаются гибких методологий разработки, но никто не использует Agile практики в чистом виде, как они описаны в стандартах. Это связано с тем, что различные типы проектов требуют различных подходов к реализации. В данном исследовании рассмотрен современный подход к реализации Fintech продуктов.

Введение. Fintech продукты направлены на улучшение и автоматизацию финансовых услуг. В большей степени fintech продукты используются, чтобы помочь компаниям, владельцам бизнеса и потребителям лучше управлять своими финансовыми операциями, процессами и жизнью, используя специализированное программное обеспечение и алгоритмы, которые используются на компьютерах и, все чаще, смартфонах.

Поскольку это направление напрямую связано с финансами, то вопрос контроля качеством таких продуктов принимает большее значение, чем для обычного IT-проекта. В данном случае любая функциональная ошибка может привести к тому что: либо компания потеряет деньги, либо пользователь потеряет деньги, а компания, соответственно, клиента.

Для того чтобы контролировать процесс разработки Fintech проектов и поддержания должного уровня качества конечного продукта используют подход, который рассмотрен далее

Основная часть. Основа рассматриваемого подхода состоит в формировании команды. Вместо чистой роли менеджера проектом используется гибридная роль QA-Lead или Developer-Lead. Преимущество новой выбранной роли в том, что такой сотрудник отлично знает предметную область и знает тонкости процессов тестирования и разработки, а также нацелен на создание качественного продукта.

Вторым шагом для данного подхода является архитектурное решение, а именно – переход на кроссплатформенную кластеризацию. Это важно для трейдинговых платформ, поскольку трейдеры используют платформу 24/7, а каждая биржа работает в разном часовом диапазоне. То есть, отличие от одно-кластерной платформы будет в том, что в случае если система будет недоступна на одном из кластеров, то она будет переключена на другой и для пользователя это переключение было незаметно. При этом ни пользователь, ни брокер не будет терпеть убыток из-за отсутствия доступа к платформе.

Аналогичные примеры используются в банковских системах, где потеря средств при переводе недопустима.

Следующим важным шагом является проведение тест-анализа еще до начала разработки программного продукта. Тестирование требований или как его еще называют «тест анализ» необходим, во-первых, для того, чтобы отсеять ошибки самих требований, т.е. устранить очевидные ошибки в требованиях, найти возможные противоречия. Во-вторых, задача тестировщика при прочтении требований, посмотреть со стороны пользователя, поэтому правильно заданный вопрос на этапе тестирования требований в перспективе позволит сэкономить деньги заказчика.

Вы должны поставить себя на место пользователя, прожить все сценарии использования, как пользователь и остальные действующие лица, оценить насколько этот функционал удобен в использовании и возможно изложить свои замечания Аналитику, но окончательное решение будет приниматься по согласованию аналитика с Заказчиком. В-третьих, тестировщик – это Вторые глаза - которые посмотрели в требования и сквозь них и предположили, что может быть упущено или противоречит каким-то стандартам.

При разработке ПО, заказчик смотрит со стороны бизнеса, то есть использует подход, ориентированный на продукт при разработке они предполагают, что функции, которые они придумали, понравятся пользователям на основании их исследований.

Часто это приводит к тому, что одни функции не используются, т.к. не нужны пользователю; другие функции пользователю очень нужны, но он не может их найти; а третьи — делают не то, что ожидается.

После завершения тест анализа, начинается разработка программного продукта и параллельно проходит тест-дизайн – написание тестовой документации. Проведенный тест-анализ позволит отделам разработки и тестирования покрыть все возможные кейсы и тем самым снизить вероятность возникновения функциональных дефектов при доставке конечного программного продукта.

Выводы. Таким образом, в современных компаниях при реализации Fintech проектов, помимо использования гибких методологий, допустимо объединять роли лида разработки и менеджера проектов или лида тестирования и менеджера проектов. Также рекомендуется уделять внимание архитектуре проекта и оценке риска отказоустойчивости на стадии планирования. И проведение процесса тест-анализа перед началом работы над разработкой программного продукта. Все эти действия в совокупности позволят повысить качество производимого продукта и упростят коммуникацию с заказчиком. Однако важно помнить, что нет идеального подхода к реализации проекта, поскольку все проекты уникальны. Поэтому тест анализ нужно проводить всегда, но объем этого процесса должен зависеть от следующих факторов и не идти в ущерб проекту:

- человеческие ресурсы;
- время, которое у нас есть для проведения тест-анализа;
- через какой промежуток времени нам нужно выпускать наш продукт.

Михальчук В.Д. (автор)

Подпись

Торосян Е.К. (научный руководитель)

Подпись