



Российские
интернет-технологии
2008

Взаимодействие отдела проектирования интерфейсов и разработчиков в agile-процессе

Юрий Ветров, Юрий Шиляев,
Александр Хмелевский

О чем доклад?

1. Контекст

Как и над чем мы работаем внутри компании?

2. Проблемы

Что ставит нам палки в колеса и мешает процессу?

3. Решения

Что позволяет нам выстраивать процесс и избегать проблем?

4. Выводы

Как выстроить процесс еще лучше?



Как и над чем мы работаем?

1. Три офиса компании — Москва, Санкт-Петербург и Минск — взаимодействуют между собой.
2. Длительные и сложные проекты — от полугода в работе и еще больше — в развитии.
3. Концепция проекта часто известна лишь в общих чертах.
4. Потребительские качества в ведущихся проектах обычно важнее функциональных.
5. Аналитики, проектировщики и дизайнеры выделены в специальный отдел.
6. Для каждого крупного проекта выделяется отдельная команда разработки.

Проблемы

1. Частое изменение требований.
2. Географическая удаленность команд и заказчика.
3. Полнота и избыточность документации.
4. Аналитики не всегда знают о нуждах команды.
5. Инструменты совместной работы.
6. Принятие решений, ответственность и полномочия.
7. Демонстрация результатов работы команды и текущего статуса проекта.



1. Частое изменение требований

- Проект разрабатывается долго и за это время может измениться рынок, а значит и требования к продукту.
- Невозможно детализировать абсолютно все в сложных проектах — только если потратить неразумно много времени на проектирование и спецификации.



2. Географическая удаленность

- Классическая проблема.
- Коммуникация затруднена — сложно оперативно решать вопросы и быстро обмениваться информацией.
- Заказчик и аккаунт-менеджер — в Москве.
- Разработчики и менеджер проекта — в Минске.
- Проектировщики — в Санкт-Петербурге.



3. Полнота документации

- Подрядчик и заказчик должны одинаково понимать, какой продукт с какими качествами получится в итоге.
- Документация должна быть достаточной, чтобы команда разработки знала что нужно делать.
- Документация не должна быть избыточной, чтобы на ее написание и поддержку не уходило слишком много времени.
- Документация должна быть хорошо организованной, чтобы разработчикам было удобно работать с ней.



4. Неясно, что нужно команде

- Проектировщики не всегда знают, над чем сейчас работает команда.
- Схемы страниц и дизайн поставляются с задержкой из-за того что проектировщики узнали о потребностях команды поздно.



5. Общие инструменты работы

- Процесс работы над проектами в отделах проектирования и разработки различается — для каждого удобен свой инструмент.
- Инструменты должны позволять совместную работу с документами, постановку и контроль выполнения задач, отчетность и систему контроля качества.

6. Принятие решений

- Не всегда ясно, кто именно отвечает за тот или иной участок работ.
- Часто непонятно, кто должен принимать решения по функциональным, потребительским и другим качествам продукта.
- Излишняя бюрократия тормозит процесс работы, снижает качество и повышает стоимость.



7. Демонстрация результатов

- Клиент хочет видеть результаты работ как можно чаще.
- Клиент хочет видеть наглядные результаты работ — картинки, картинки и еще раз картинки.
- Клиенту важно знать, что сейчас делается и сделано ли то, что уже запланировано.



Решения

1. Гибкие методики ведения проектов.
2. Командировки и конференции.
3. Четко выстроенный процесс проектирования.
4. Планирование итераций заранее.
5. Использование общих менеджеров задач и других систем.
6. Четкая схема принятия решений, передача многих полномочий и ответственности разработчикам.
7. Регулярные презентации заказчику.



~~1. Частое изменение требований~~

- Переход к итеративному процессу разработки, когда продукт обновляется небольшими порциями раз в 1-2 недели.
- Использование *agile-практик* ведения проектов, которые делают процесс ведения проекта более прозрачным и контролируемым.



~~2. Географическая удаленность~~

- Не решаема, но и не смертельна.
- Заказчик, разработчики и проектировщики находятся в пределах пары часов полета или ночи на поезде.
- Сами команды работают вместе и не разделены — проектировщики работают с проектировщиками, разработчики — с другими разработчиками, тестировщиками и менеджером.
- Аккаунт-менеджер находится в том же городе, что и клиент.

~~2. Географическая удаленность~~





~~3. Полнота документации~~

- Четко отработан состав документации и процесс работы над ней.
- Со временем отказались от громоздких документов и тех, которые слишком сложно поддерживать.
- Сперва прорабатывается и визуализируется в виде интерактивного прототипа концепция продукта. Прототип — часть документации.
- Вначале проектируются крупные процессы, а уже более мелкие вещи — по мере необходимости. Принцип «Just in Time» — это и скорость, и качество работ, и лучшее планирование.

~~3. Полнота документации~~



Бизнес-анализ

- Видение
- Описание целевой аудитории
- Сценарии взаимодействия
- Перечень функциональности (*user stories*)
- Критерии приемки



Проектирование

- Карта сайта
- Диаграммы взаимодействия
- Структурные схемы страниц (*wireframes*)



Дизайн

- Дизайн-макеты ключевых страниц
- Типовые элементы интерфейса



Прототип

- Шаблоны страниц
- Сборка прототипа и наполнение контентом



~~4. Неясно, что нужно команде~~

- Работы по проектированию и дизайну планируются заранее — как минимум на одну итерацию.
- Инициирование общения от разработчиков — они лучше всего знают, что им нужно для работы.
- Участие проектировщиков в удаленных митингах позволяет лучше понимать проблемы разработчиков и знать о том, что они собираются делать в ближайшее время.

~~4. Неявно, что нужно команде~~



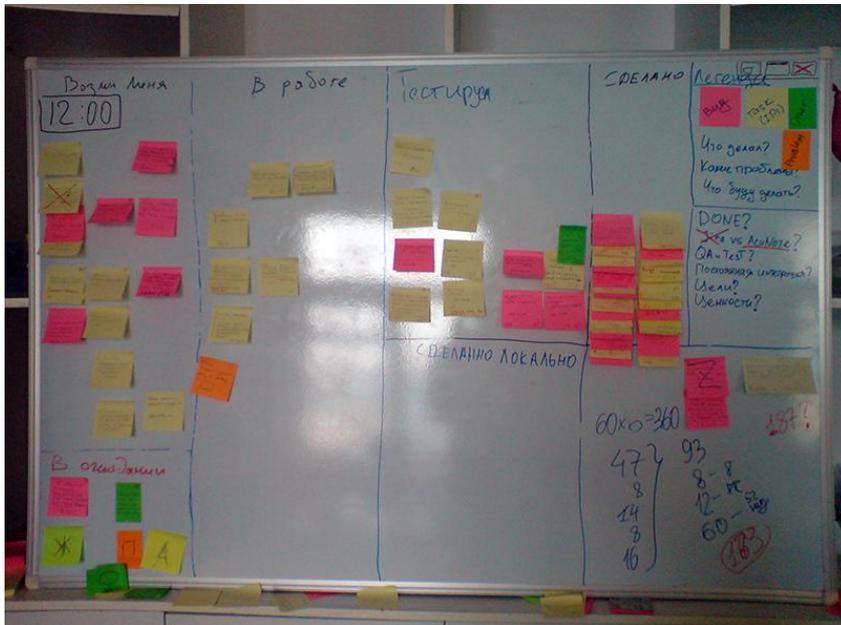
Итерация №1
Проектирование модуля №2
Разработка модуля №1

Итерация №2
Проектирование модуля №3
Разработка модуля №2

~~5. Общие инструменты работы~~

- Команда активно использует *offline* инструменты:
 - **Taskboard** — для постановки задач.
 - **Маркерные доски** и флип-чарты — для планерок и митингов.
- Внедрен единый менеджер задач — онлайн-сервис «Acunote».
- Используется система баг-трекинга.
- Все документы, файлы и код проходят через систему контроля версий.

5. Общие инструменты работы

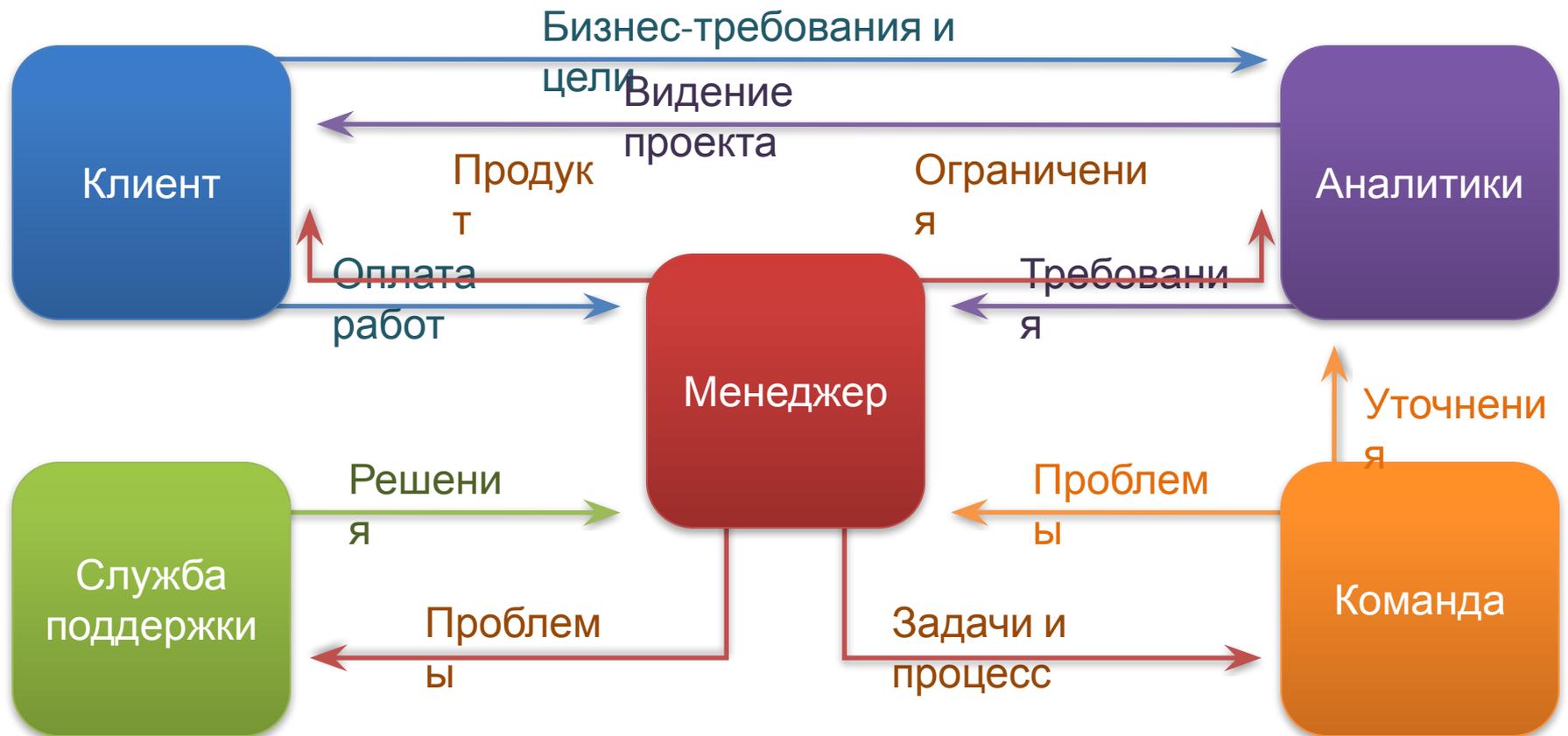


New Task		Move	Copy	Go to	Backlog	Delete	Reorder
Description	Owner	Status	Est.	Rem.	filtered by: All Users All Tasks		
<input type="checkbox"/> <input type="checkbox"/> Invent premium features for first 100 clients	bill	Deferred	2	0			
<input type="checkbox"/> <input type="checkbox"/> setup https://secure.ourproject.com	ted	In Progress	6	0.5			
<input type="checkbox"/> <input type="checkbox"/> purchase SSL certificate	ted	Completed	4	0			
<input type="checkbox"/> <input type="checkbox"/> configure SSL host in Apache	ted	In Progress	2	0.5			
<input type="checkbox"/> <input type="checkbox"/> switch users to Free Plan when payment transaction fails	bill	In Progress	5	1			
<input type="checkbox"/> <input type="checkbox"/> email notifications for clients on this	bill	Not Started	1	1			
<input type="checkbox"/> <input type="checkbox"/> switch premium features on and off when Pricing Plan changes	bill	Completed	4	0			
<input type="checkbox"/> <input type="checkbox"/> implement credit-card validity-checking	joe	In Progress	1	0.5			

~~6. Принятие решений~~

- Четко очерчены сферы ответственности каждого участника проекта — кто, когда и за какие качества проекта отвечает.
- Переход от функционального разделения труда к разделению по участкам работы.
- Разработчики должны иметь достаточные полномочия для принятия решений на своем участке работ, чтобы не бегать каждый раз к менеджеру.
- Все ответственны за все. Это не означает безответственность, если менеджер проекта следит за общим процессом и является его модератором.

6. Принятие решений





~~7. Демонстрация результатов~~

- Прозрачность перед клиентом:
 - открытый клиенту таск-менеджер и баг-трекер;
 - демо-сервер, на котором можно увидеть и протестировать следующий релиз продукта.
- Демонстрация результатов по всем важным вехам у клиента.
- Регулярная отчетность благодаря частым итерациям.
- Аккаунт-менеджер присутствует даже на внутренних совещаниях заказчика.
- Демонстрация картинок и интерактивных прототипов как можно чаще и как можно раньше.

~~7. Демонстрация результатов~~

1

Прозрачность

- Таск-менеджер
- Баг-трекер
- Демо-сервер

2

Демонстрация результатов

- Презентация вех
- Интерактивный прототип и дизайн
- Регулярная отчетность



Выводы

1. Продолжаем внедрение гибких методик разработки.
2. Ищем баланс между чистыми концепциями *agile* и *user-centered design*.
3. Работаем над скоростью работы отдела проектирования.



1. Дальнейшее внедрение agile

- Процесс внедрения *agile* небыстрый и непростой — нужно здорово сдвинуть точку сборки у всей проектной команды. Зато эффект внедрения здорово повысит эффективность.
- Сложно убедить заказчика в том, что *agile* — это хорошо и удобно для обеих сторон. Но после этого процесс станет более выгодным и комфортным для обеих.
- Полномочия и ответственность в команде иногда приходится насаждать почти насильно. Но без этого невозможна ни успешная команда в целом, ни профессионал в отдельности.

2. Баланс между agile и UCD

- *Agile* предполагает как можно более ранний старт разработки. Но не всегда известна концепция проекта, чтобы можно было начинать работы — нужно сперва проработать ее.
- *User-Centered Design* предполагает как можно большую проработку интерфейса перед началом разработки. Но не всегда есть смысл продумывать все мелочи заранее. Работы разбиваются на два или три этапа, в зависимости от проекта: проработка концепции, проектирование основных процессов и детальное проектирование интерфейса.



3. Ускорение проектирования

- Перенос части работ по проектированию из предварительных работ в поддержку — проектирование функций делается по запросу команды.
- Автоматизация части работ — ускорение отрисовки схем страниц, использование CSS-фреймворков для сборки интерактивного прототипа.
- Стандартизация документов и процесса проектирования. Скорость работы отдела важна как для команды разработки, так и для быстрой оборачиваемости в самом отделе.



Спасибо!

- Юрий Ветров, руководитель отдела проектирования
www.jvetrau.com
- Юрий Шильяев, руководитель офиса разработчиков
yuri.shilyaev.com
- Александр Хмелевский, проектировщик интерфейсов
www.axme.ru



UI MODELING COMPANY
Проектирование пользовательских интерфейсов

www.uimodeling.ru



artics
internet solutions

www.artics.ru