## Что необходимо сделать

* 1. Сравните между собой все изученные методологии. Сделайте таблицу-сравнение этих методологий.
  2. Определите, какая из методологий, по вашему мнению, больше всего подойдет для решения поставленной задачи и опишите, почему.
  3. Подготовьте презентацию, которая поможет убедить руководство прислушаться к вам и применить современные подходы разработки в новом проекте. Редактор для создания презентации остается на ваше усмотрение.
  4. Прикрепите ссылку на презентацию и отправьте на проверку ментору. Важно проверить, чтобы доступ по ссылке был открыт.
  5. Подготовьте черновик плана по внедрению выбранной вами методологии.
  6. Добавьте все ссылки: на таблицу, презентацию, черновик плана по внедрению выбранной методологии и описание, почему именно ее стоит выбрать, в окошко для сдачи задания на проверку ментору.

|  |  |  |
| --- | --- | --- |
| Методология | Плюсы | Минусы |
| Scrum | * это сотрудничество: команда работает вместе, чтобы определить цели спринта и достигнуть целей; * помогает командам брать большие проекты и разбивать их на более мелкие, управляемые куски; * обратная связь непрерывна, и команда включает ее в следующую версию; * в меньшие разделы легче вносить изменения; * работа в небольших спринтах повышает вероятность того, что команда поймает ошибки перед выпуском. | * Scrum может быть совершенно новым способом ведения дел, и некоторым сложно адаптироваться; * все должны быть командными игроками, а это не всегда так; * если член команды выходит из середины спринта, может быть трудно закончить вовремя; * не все любят ежедневные встречи. |
| Kanban | 1. Гибкость планирования. Команда концентрируется только на текущей работе, приоритет задачи выставляется менеджером. 2. Высокое вовлечение команды в процесс разработки. Благодаря постоянным собраниям, прозрачности процессов и возможностям самоорганизации работники сплачиваются и проявляют искренний интерес. 3. Меньшая продолжительность цикла. Если несколько человек обладает схожими навыками, продолжительность сокращается, если же только один — появляется узкое место. Поэтому сотрудники должны делиться знаниями и тем самым оптимизировать продолжительность цикла. Тогда вся команда сможет взяться за работу, которую забуксовала,и восстановить плавный поток. 4. Меньше узких мест. Лимиты РВП позволяют быстро находить узкие и проблемные места, которые появились из-за дефицита внимания, людей или навыков. 5. Наглядность. Когда все исполнители имеют доступ к данным, то узкие места легче заметить. Kanban-команды, помимо самих карточек, обычно используют два общих отчета: графики управления и совокупного потока. | 1. система плохо работает с командами численностью более 5 человек 2. он не предназначен для долгосрочного планирования. |

Существует еще ряд Agile методологий:

* экстремальное программирование (Extreme Programming, XP);
* разработку, управляемую функциональностью (Feature-driven development, FDD);
* разработку через тестирование (Test-driven development, TDD);
* методологию «чистой комнаты» (Cleanroom Software Engineering);
* итеративно-инкрементальный метод разработки (OpenUP);
* методологию разработки Microsoft Solutions Framework (MSF);
* метод разработки динамических систем (Dynamic Systems Development Method, DSDM);

Но для условий стартапа, как мне кажется, они не подходят.

Для использования в данном проекте на этапе разработки я предлагаю использовать Scrum. Scrum делит проект на части, которые заказчик может сразу использовать для получения ценности. Далее владелец продукта, то есть представитель заказчика в команде, определяет приоритетные составляющие проекта. Самые важные из них должны быть первыми выполнены в спринте. По итогам спринта заказчик получает рабочий инкремент продукта, то есть элементы, которыми уже можно пользоваться.

Что значит, что наш заказчик сможет быстро выйти на рынок.

При условии, что я погружен в область довольно глубоко или команда наймет Scrum-тренера, план внедрения может выглядеть следующим образом:

### 1. Анализ ситуации и выбор процесса/консультации с руководством

### Цели:

- заручиться согласием менеджмента на изменения;

Процесс:

Первое, что нужно сделать, внедряя новый процесс это добиться принятия программы изменений менеджментом. Для начала надо составить общее видение текущей ситуации у всех заинтересованных лиц - заказчиков внедрения, руководителей проектов, разработчиков. Какими вещами очень довольны, какие хотелось бы изменить. Какие может быть даже на первый взгляд нереальные цели хотелось бы достичь.

### 2. Базовый курс Scrum для команды

### Цели:

- обеспечить понимание командой базовых принципов Agile;

- обнаружить подводные камни, специфические для ваших проектов;

- выработать общее видение изменений и составить план перехода;

- определить энтузиастов Agile-методов, которые смогут стать в дальнейшем командой внедрения;

- настроить коллектив на изменения;

- составить план перехода и график тренингов;

Процесс:

Однодневный "базовый" командный тренинг, объясняющий базовые принципы Agile и основы процесса. После этого мы составляем список необходимых преобразований и обсуждаем возможные последствия с каждым членом команды. За счёт обратной связи удаётся настроить процесс оптимальным образом, а также создать благоприятный для изменений климат в коллективе. По итогам обсуждений мы составляем документ, описывающий новый процесс, его преимущества, и причины перехода со старого процесса, а также график тренингов.

### 3. Адаптация Scrum в команде

### Цели:

- обеспечить готовность к запуску первой итерации.

Процесс:

Двухдневный "продвинутый" командный тренинг, описывающий особенности реализации процесса в команде. Проводятся тренировки по таким аспектам Scrum как:

- планирование релиза

- планирование итерации

- совещание-летучка

- ретроспектива

- демонстрация

а также, как правило, вводный курс по сопутствующим инженерным практикам:

- agile version control

- постоянная интеграция

- test-driven development

Дополнительные тренинги для ролей Scrum-мастера (4 часа) и Product Owner'а (4 часа).

### 4. Пробная итерация

### Цели:

- тест-драйв.

Процесс:

После тренинга начинается первая итерация. Планирование итерации происходит в присутствии тренера. тренер отвечает на возникающие вопросы и помогает модерировать дискуссию.

В процессе итерации тренер несколько раз присутствует на stand-up митингах (летучках), корректируя возникающие ошибки.

Первая ретроспектива в конце итерации и демонстрация продукта также происходит с его участием.

Длительность: две-три недели (в зависимости от выбранной длительности итерации)

### Далее

А далее команда работает по Scrum самостоятельно, тренер участвует всё меньше и меньше.

## Резюме

Суммарная длительность внедрения: 6-8 недель.

Результаты:

- достингута прозрачность процесса для руководства и команды;

- понятна производительность команды и более предсказуемы сроки релизов (пока ещё ближайших);

- определены пути и конкретные шаги по увеличению производительности и уменьшению количества дефектов;

- мотивация команды существенно повысилась;