**Beet Seed**

| **№** | **Название методологии** | **Сильные стороны** | **Слабые стороны** | **В какой области целесобразно применять** |
| --- | --- | --- | --- | --- |
| 1 | Waterfall (каскадная модель) | 1. Данная методология является последовательной. Все процессы структурированы и позволяют планировать деятельность на каждом этапе. Я считаю это полезным, т.к. каждый этап понятен и позволяет сфокусироваться только на нём.  2. Отсюда и “вытекает” прогнозируемость. То есть, каждый этап может быть спланирован на основе предыдущего с чёткими сроками перехода на другой.  Я считаю, что имея определённую информация, которая тщательно сформированная на предыдущем этапе позволяет правильно планировать текущий.  3. Завершенность. Последующие переходы с одного этапа на другой подразумевают под собой окончание предыдущего этапа, то есть последующие не начнутся без финализации предыдущего.  Данный пункт является положительным, т.к. не позволяет начаться новому этапу без тчательной проработки предыдущего, что в свою очередь минимизирует риски. | 1. Данная модель не является гидбой. Все этапы чётко фиксированные, то есть продукт проходит через каждый этап разработки без возврата на предыдущий.  По моему мнению, это является минусом, т.к. есть вероятность ошибок на прошлых этапах и соотвественно нужна неоходимость вовзращаться назад для их исправления, что не вписывается в идеологию данной концепции.  2. Данная модель не позволяет “пощупать” продукт в готовом состоянии без выполнения всех этапов разработки.  Я думаю, что в наше время очень важно иметь готовый продукт на ранних этапах, чтобы провести исследования или опрос фокус-группы, то есть важна скорость и доступность, а данный пункт является минусом, т.к. не подразумевает под собой готовой продукции.  3. Очень рискованая и дорогостоящая. Если были допущены ошибки на ранних этапах разработки, то цена их исправления как временная так материальная очень высока.  Моё мнение таково, что если продукт имеет проблемы, который были допущены на ранних этапах разработки, то стоимость их устранения очень велика, что в свою очередь сказывается на доходах, репутации и т.д.. | Обычно применяется на сложных и дорогостоящих проэктах таких как автомобилестроение, космическая и строительные области, медицина и т.д.  Моё мнение:  Ранее, в 20 веке, когда был индустриальный “бум”, были сформированы принципы и структура, основу которой, заложили инженеры.  Массовое производство продукции подразумевало под собой налаживания технологических процессов производимых на конвеерах, поэтому необходимо было тщательно планировать каждый этап, чтобы в ближайшие сроки не вмешиваться в произовдство, так как это очень было дорого и затратно по времени. |
| 2 | Scrum (итарационная модель) | 1. Данная модель является гибкой и подразумевает под собой возмодность направлять вектор разработки, т.е. команда может менять требования, подходы, сроки и т.д. в зависимости от внешних факторов. Я считаю это плюсом, так как различные факторы могут меняться в процессе разработки, которые могут вносить свои коррективы и команда должна быть готова к этому.  2. Участие различных специалистов на каждом этапе разработки. Я думаю, чтобы создать отличный продукт, необходимо услышать мнение всех специалистов, которые работают над разработкой, так как один отдел не может предугадать различные ботл-неки. Из этого следует, что снижаются риски возникновения проблем в будущем.  3. Получение продукта, пусть и в базовом варианте в кратчайший срок. Я думаю, что данный пункт можно отнести к позитивным моментам, чтобы позволить заинтересованному кругу лиц увидеть результаты работы и на основнаии этого начать получать выгоду, а также внести какие либо корректировки. | 1. Не подойдет для компаний, которые имеют чёткие сроки и ограниченный бюджет. По сути компания имеет определённо сформулированную цель и средства, которые не должны меняться в процессе разработки. Как мне кажеться, это является минусом, что в данном случае методология не подходит.  2. В связи с частыми итерациями, изначальные цели могут быть потерены, что в своб очередь может сместить фокус не в ту сторону. Как мне кажется, есть велика вероятность с этой методологией и постоянной лояльностью к изменениям уйти “не в ту степь”.  3. Требуется высокая вовлеченность команды. То есть, команда должна быть довольно опытна, должна активно принимать участие в разработке, отсюда следует, что знания и навыки специалистов должны быть на высоком уровне, чтобы следовать этой методологии.  По моему мнению, в связи с быстро-меняющимеся “вводными” может быть сложно “подстроиться” команде или людям в определённый процесс, что может быть минусом для тех людей, которым сложно быть гибким, отсюда следует проблема мотивации и далее результативности. | Целесообразен в таких сферах как разработка програмного обеспечения, стартапы, маркетинг, продуктовое развитие.  Моё мнение такого: везде где есть взаимосвязь клиента с продуктом, где необходима гибкость, а в наше время, где повсеместная конкуренция и условия рынка всё время меняются необходимо иметь возможность подстраивать свои продукты под реалии. Также важно улучшать продукт в кратчайший срок, чтобы иметь возможность что-то добавлять/удалять и быть “на острие ножа”. |

**Beet Sprout**

Agile манифест появился в начале 21 века, в то время когда IT индустрия стала очень быстро развиваться, но в том время преобладала каскадная модель. Особенности того времени подразумевали под собой иметь возможность создавать инновации и люди стали понимать, что в привычном понимании каскадная модель не предоставляет возможности быстро реагировать не зменчивые условия рынка. Собравшись вместе специалисты из разных областей разработки и продвижений прешли к выводу, что необходимо менять подходы, так как усторевшие методы уже не работают. Плюс, как я думаю, руководети производств и большие корпорации, зачастую, неспособны (а скорее не имеют желания) к резким изменениям, так как это влечёт за собой временные и финансовые затраты. Но так как время идёт, меняется рынок, конкуренция растёт, а воможно и условия труда являются не подобающими, то возникла идея, как бы изменить подходы в производстве и изенения процессов. Основные причины, которые привели к появлению Agile манифеста: несоотвествие реальности (то как было заложенно и то как стало в то время), низкая гибкость, высокие риски, низкая удовлетворённость клиентов.

Как и свод правил, а манифест по сути им и является, основан на определённых постулатах, которые в свою очередь должны поднять процессы на высший уровень.

Люди и взаимодействие важнее процессов и инструментов. Является основополагающим, так как люди являются ядром всего, процессов, продуктов и взаимоотношений, поэтому во главу ставят именно людей.

Работающий продукт важнее исчерпывающей документации. Этот пункт является следующим, так как подразумевает, что важнее получить результат работы, который будет приносить пользу и доход, а не зацикленность на документах, которые могут отнимать дополнительное время и средства.

Сотрудничество с заказчиком важнее согласования условий контракта. Данный аспект показывает насколько важны взаимоотношения и тесная связь с клиентом/заказчиком, чем следованию правил и чётких договорённостей, чем в свою очередь и является гибкость.

Готовность к изменениям важнее следования первоначальному плану. Иметь возможность менять подстраиваться и корректировать планы также является очень важным, так как в процессе разработки могут появлятся различные факторы, которые не могли быть спрогнозированны, соотвественно способность к гибкости является обязательным фактором, который и включён в этот пункт.

Таким образом, Agile-манифест стал ответом на потребность в более гибком и адаптивном подходе к разработке программного обеспечения, который позволяет создавать качественные продукты, удовлетворяющие потребности клиентов.

Agile-манифест был ответом на ряд проблем, которые существовали в традиционной, каскадной модели разработки программного обеспечения. Основными задачами, которые ставили перед собой авторы манифеста, были:

Улучшение гибкости разработки, повышение качества продукта, улучшение взаимодействия с заказчиком, повышение удовлетворенности команды

Можно сказать, что Agile-манифест в значительной степени достиг своих целей. Agile-методологии стали одними из самых популярных в мире разработки программного обеспечения, соответственно они стали широкораспространёнными. Многие компании, применяющие Agile, отмечают повышение качества своих продуктов. Agile-проекты, как правило, позволяют быстрее и эффективнее реагировать на меняющиеся потребности клиентов. Agile-команды, как правило, более продуктивны и удовлетворены своей работой.

В целом, Agile-манифест стал важным этапом в развитии методологий разработки программного обеспечения. Он позволил преодолеть многие ограничения традиционных моделей и предложил более гибкий и эффективный подход. Однако, как и любой другой подход, Agile имеет свои сильные и слабые стороны и требует взвешенного подхода при выборе и применении.

**Mighty Beet**

В данном случае я бы воспользовался итеративной моделью разработки (возможно часть скрама или кандана “Скрамбан”). Так как она является оптимальной при разработке програмного продукта. Данная методология позволит мне правильно и эффективно построить процесс разработки, а также иметь возможность подстариваться под реальности и в случае чего изменять вектор развития моего продукта. Также мне важно, как владельцу проекта непосредственно влиять на все этапы разработки, получать фитбеки от клиентов, чтобы улучшать мой продукт. От команды я ожидаю полной организованности и вовлечённости в процесс. Возможность выпуска продукта или его части является очень важной, так как позволяет быстро реагировать на проблемы, а так же доставлять новый функционал в кратчайший срок. В процессе разработки важно понимать отслеживаемость и прозрачность, а эта методология по общим показателям позволяет этого достичь. Наличие доски с задачами и распределение между специалистами разных отделов позволяет наблюдать и контролировать все процессы этапов разработки. Наличие коммуникации на всех этапах разработки является также важным, так как можно понимать состояния продукта в определённом промежутке времени. В целом, для данного продукта, итеративная модель разработки является правильной и полезной так как включает в себя много плюсов.