



0

Общий рейтинг

Edison

Изобретаем успех: софт и стартапы

[Подписаться](#)

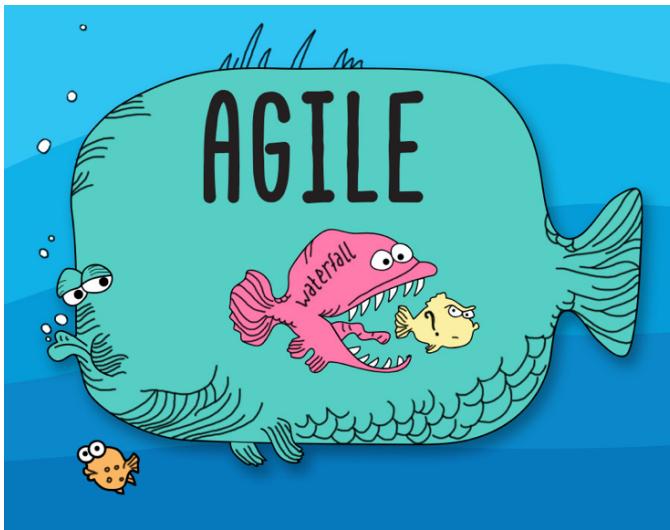
ThomasAlva 3 ноя 2015 в 20:20

Ещё раз про семь основных методологий разработки

8 мин 1.1M

Блог компании Edison, Программирование*,
Веб-разработка*,
Разработка мобильных приложений*

Разработка программного продукта знает много достойных методологий — иначе говоря, устоявшихся best practices. Выбор зависит от специфики проекта, системы бюджетирования, субъективных предпочтений и даже темперамента руководителя. В статье описаны методологии, с которыми мы регулярно сталкиваемся в Эдисоне.



1. «Waterfall Model» (каскадная модель или «водопад»)



Одна из самых старых, подразумевает последовательное прохождение стадий, каждая из которых должна завершиться полностью до начала следующей. В модели Waterfall легко управлять проектом. Благодаря её жесткости, разработка проходит быстро, стоимость и срок заранее определены. Но это палка о двух концах. Каскадная модель будет давать отличный результат только в проектах с [четко и заранее определенными требованиями](#) и

способами их реализации. Нет возможности сделать шаг назад, тестирование начинается только после того, как разработка завершена или почти завершена. Продукты, разработанные по данной модели без обоснованного ее выбора, могут иметь недочеты (список требований нельзя скорректировать в любой момент), о которых становится известно лишь в конце из-за строгой последовательности действий.

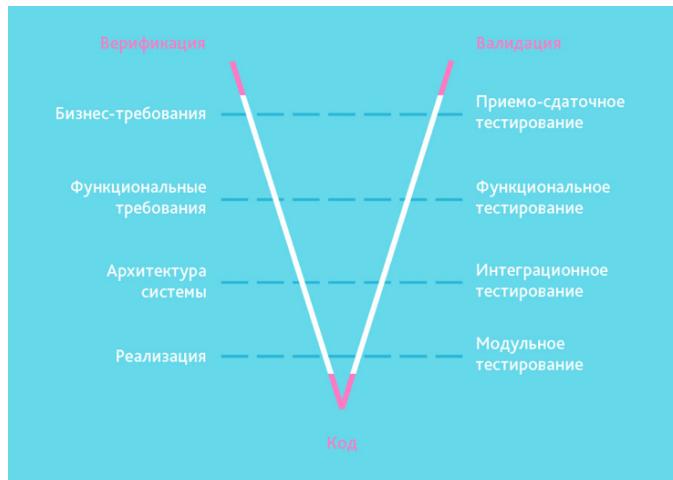
Стоимость внесения изменений высока, так как для ее инициализации приходится ждать завершения всего проекта. Тем не менее, фиксированная стоимость часто перевешивает минусы подхода. Исправление осознанных в процессе создания недостатков возможно, и, по нашему опыту, требует от одного до трех дополнительных соглашений к контракту с небольшим ТЗ.

С помощью каскадной модели мы создали множество проектов «с нуля», включая разработку только ТЗ. Проекты, о которых написано на Хабре: средний — [рентгеновский микротомограф](#), мелкий — [автообновление службы Windows на AWS](#).

Когда использовать каскадную методологию?

- Только тогда, когда требования известны, понятны и зафиксированы. Противоречивых требований не имеется.
- Нет проблем с доступностью программистов нужной квалификации.
- В относительно небольших проектах.

2. «V-Model»



Унаследовала структуру «шаг за шагом» от каскадной модели. V-образная модель применима к системам, которым особенно важно бесперебойное функционирование. Например, прикладные программы в клиниках для наблюдения за пациентами, интегрированное ПО для механизмов управления аварийными подушками безопасности в транспортных средствах и так далее. Особенностью модели

можно считать то, что она направлена на тщательную проверку и [тестирование продукта](#), находящегося уже на первоначальных стадиях проектирования. Стадия тестирования проводится одновременно с соответствующей стадией разработки, например, во время кодирования пишутся модульные тесты.

Пример нашей работы на основе V-методологии — мобильное приложение для европейского сотового оператора, который экономит расходы на роуминг во время путешествий. Проект выполняется по четкому ТЗ, но в него включен значительный этап тестирования: удобства интерфейса, функционального, нагрузочного и в том числе интеграционного, которое должно подтверждать, что несколько компонентов от различных производителей вместе работают стablyно, невозможна кража денег и кредитов.

Когда использовать V-модель?

- Если требуется тщательное тестирование продукта, то V-модель оправдает заложенную в себя идею: validation and verification.
- Для малых и средних проектов, где требования четко определены и

фиксированы.

- В условиях доступности инженеров необходимой квалификации, особенно тестировщиков.

3. «**Incremental Model**» (инкрементная модель)

В инкрементной модели полные требования к системе делятся на различные сборки. Терминология часто используется для описания поэтапной сборки ПО. Имеют место несколько циклов разработки, и вместе они составляют жизненный цикл «мульти-водопад». Цикл разделен на более мелкие легко создаваемые модули. Каждый модуль проходит через фазы определения требований, проектирования, кодирования, внедрения и тестирования. Процедура разработки по инкрементной модели предполагает выпуск на первом большом этапе продукта в базовой функциональности, а затем уже последовательное добавление новых функций, так называемых «инкрементов». Процесс продолжается до тех пор, пока не будет создана полная система.



Инкрементные модели используются там, где отдельные запросы на изменение ясны, могут быть легко формализованы и реализованы. В наших проектах мы применяли ее для создания читалки DefView, а следом и сети электронных библиотек Vivaldi.

Как пример опишем суть одного инкремента. Сеть электронных библиотек Vivaldi пришла на смену DefView. DefView подключалась к одному серверу документов, а теперь может подключаться ко многим. На площадку учреждения, желающего транслировать свой контент определенной аудитории, устанавливается сервер хранения, который напрямую обращается к документам и преобразует их в нужный формат. Появился корневой элемент архитектуры — центральный сервер Vivaldi, выступающий в роли единой поисковой системы по всем серверам хранения, установленным в различных учреждениях.

Когда использовать инкрементную модель?

- Когда основные требования к системе четко определены и понятны. В то же время некоторые детали могут дорабатываться с течением времени.
- Требуется ранний вывод продукта на рынок.
- Есть несколько рисковых фич или целей.

4. «RAD Model» (rapid application development model или быстрая разработка приложений)

RAD-модель — разновидность инкрементной модели. В RAD-модели компоненты или функции разрабатываются несколькими высококвалифицированными командами параллельно, будто несколько мини-проектов. Временные рамки одного цикла жестко ограничены. Созданные модули затем интегрируются в один рабочий прототип. Синергия позволяет очень быстро предоставить клиенту для обозрения что-то рабочее с целью получения обратной связи и внесения изменений.



Модель быстрой разработки приложений включает следующие фазы:

- Бизнес-моделирование:
определение списка информационных потоков между различными подразделениями.
- Моделирование данных:
информация, собранная на предыдущем этапе, используется для определения объектов и иных сущностей, необходимых для циркуляции информации.
- Моделирование процесса:
информационные потоки связывают объекты для достижения целей разработки.
- Сборка приложения: используются средства автоматической сборки для преобразования моделей системы автоматического проектирования в код.
- Тестирование: тестируются новые

ИНФОРМАЦИЯ

Сайт www.edsd.ru

Дата регистрации 14 сентября 2015

Дата основания 16 января 2002

Численность 31–50 человек

Местоположение Россия

компоненты и интерфейсы.

Когда используется RAD-модель?

Может использоваться только при наличии высококвалифицированных и

Все потоки

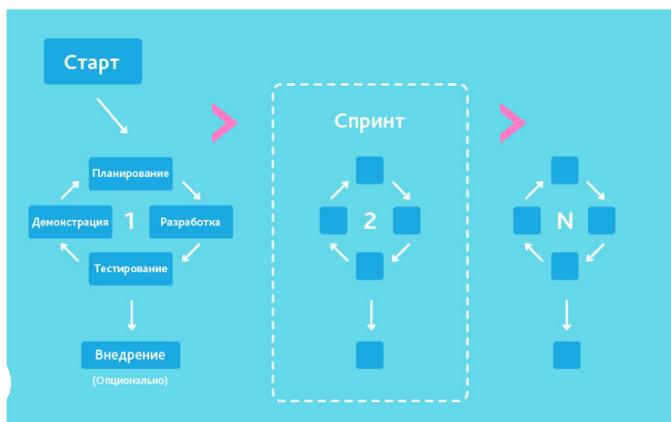


Качайте свои скиллы

[Войти](#)

стоимостью готовых инструментов автоматизированной сборки. RAD-модель может быть выбрана при уверенном знании целевого бизнеса и необходимости срочного производства системы в течение 2-3 месяцев.

5. «Agile Model» (гибкая методология разработки)



В «гибкой» методологии разработки после каждой итерации заказчик может наблюдать результат и понимать, удовлетворяет он его или нет. Это одно из преимуществ гибкой модели. К ее недостаткам относят то, что из-за

БЛОГ НА ХАБРЕ

3 июля 2020 в 21:40

Нет Cookies, нет проблем — использование ETag для отслеживания

2 июля 2020 в 23:44

Восходящая сортировка кучей

9.6K 6 +6

28 июня 2020 в 23:59

Турнирная сортировка

19K 1 +1

18 июня 2020 в 15:36

Определяем выигрышную покерную руку с помощью map/reduce на JavaScript

11K 9 +9

14 июня 2020 в 22:12

Пока, Python. Привет, Julia !

24K 71 +71

ИСТОРИИ

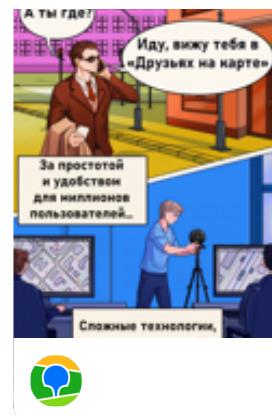


отсутствия конкретных формулировок результатов сложно оценить трудозатраты и стоимость, требуемые на разработку. Экстремальное программирование (XP) является одним из наиболее известных применений гибкой модели на практике.

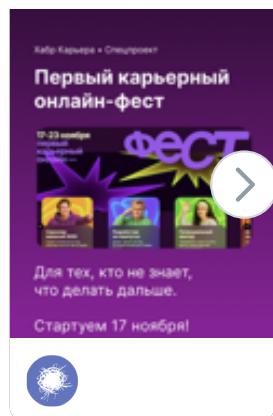
В основе такого типа — непродолжительные ежедневные встречи — «Scrum» и регулярно повторяющиеся собрания (раз в неделю, раз в две недели или раз в месяц), которые называются «Sprint». На ежедневных совещаниях участники команды обсуждают:

- отчёт о проделанной работе с момента последнего Scrum'a;
- список задач, которые сотрудник должен выполнить до следующего собрания;
- затруднения, возникшие в ходе работы.

Методология подходит для больших или нацеленных на длительный жизненный цикл проектов, постоянно адаптируемых к условиям рынка. Соответственно, в процессе реализации требования изменяются. Стоит вспомнить класс творческих людей, которым свойственно генерировать, выдавать и



Как устроен мир по ту сторону карты



Первый карьерный онлайн-фест

опробовать новые идеи еженедельно или даже ежедневно. Гибкая разработка лучше всего подходит для этого психотипа руководителей. Внутренние стартапы компаний мы разрабатываем по Agile. Примером клиентских проектов является [Электронная Система Медицинских Осмотров](#), созданная для проведения массовых медосмотров в считанные минуты. Во втором абзаце [этого отзыва](#), наши американские партнеры описали очень важную вещь, принципиальную для успеха на Agile.

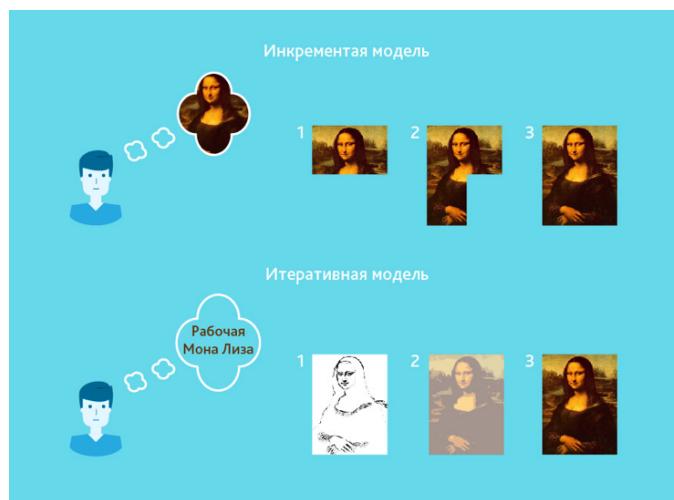
Когда использовать Agile?

- Когда потребности пользователей постоянно меняются в динамическом бизнесе.
- Изменения на Agile реализуются за меньшую цену из-за частых инкрементов.
- В отличие от модели водопада, в гибкой модели для старта проекта достаточно лишь небольшого планирования.

6. «Iterative Model» (итеративная или итерационная модель)

Итерационная модель жизненного цикла не требует для начала полной спецификации требований. Вместо

этого, создание начинается с реализации части функционала, становящейся базой для определения дальнейших требований. Этот процесс повторяется. Версия может быть неидеальна, главное, чтобы она работала. Понимая конечную цель, мы стремимся к ней так, чтобы каждый шаг был результативен, а каждая версия — работоспособна.



На диаграмме показана итерационная «разработка» Мона Лизы. Как видно, в первой итерации есть лишь набросок Джоконды, во второй — появляются цвета, а третья итерация добавляет деталей, насыщенности и завершает процесс. В инкрементной же модели функционал продукта наращивается по кусочкам, продукт составляется из частей. В отличие от итерационной модели, каждый кусочек представляет собой целостный элемент.

Примером итерационной разработки

может служить распознавание голоса.
Первые исследования и подготовка научного аппарата начались давно, в начале — в мыслях, затем — на бумаге. С каждой новой итерацией качество распознавания улучшалось. Тем не менее, идеальное распознавание еще не достигнуто, следовательно, задача еще не решена полностью.

Когда оптимально использовать итеративную модель?

- Требования к конечной системе заранее четко определены и понятны.
- Проект большой или очень большой.
- Основная задача должна быть определена, но детали реализации могут эволюционировать с течением времени.

7. «Spiral Model» (спиральная модель)



«Сpirальная модель» похожа на инкрементную, но с акцентом на анализ рисков. Она хорошо работает для решения критически важных бизнес-задач, когда неудача несовместима с деятельностью компании, в условиях выпуска новых продуктовых линеек, при необходимости научных исследований и практической апробации.

Сpirальная модель предполагает 4 этапа для каждого витка:

1. планирование;
2. анализ рисков;
3. конструирование;
4. оценка результата и при удовлетворительном качестве переход к новому витку.

Эта модель не подойдет для малых проектов, она резонна для сложных и дорогих, например, таких, как разработка системы документооборота для банка, когда каждый следующий шаг требует большего анализа для оценки последствий, чем программирование. На проекте по [разработке СЭД для ОДУ Сибири СО ЕЭС](#) два совещания об изменении кодификации разделов электронного архива занимают в 10 раз больше времени, чем объединение двух

папок программистом. Государственные проекты, в которых мы участвовали, начинались с подготовки экспертным сообществом дорогостоящей концепции, которая отнюдь не всегда бесполезна, поскольку окупается в масштабах страны.

Подытожим

На слайде продемонстрированы различия двух наиболее распространенных методологий.

В современной практике модели разработки программного обеспечения многовариантны. Нет единственно верной для всех проектов, стартовых условий и моделей оплаты. Даже столь любимая всеми нами Agile не может применяться повсеместно из-за неготовности некоторых заказчиков или невозможности гибкого финансирования. Методологии частично пересекаются в средствах и отчасти похожи друг на друга. Некоторые другие концепции использовались лишь для пропаганды собственных компиляторов и не привносили в практику ничего нового.

О технологиях разработки:

[Ещё раз про семь основных](#)

[методологий разработки.](#)

[10 главных ошибок масштабирования систем.](#)

[8 принципов планирования разработки, упрощающих жизнь.](#)

[5 главных рисков при заказной разработке ПО.](#)

Только зарегистрированные пользователи могут участвовать в опросе. [Войдите](#), пожалуйста.

Какие методологии используете вы?

20.01% Waterfall 534

56.3% Agile 1502

2.47% V-Model 66

11.32% Iterative 302

9.03% Incremental 241

2.32% RAD 62

3.79% Spiral 101

1.65% Другая (расскажу в комментариях) 44

25.71% Посмотреть ответы 686

Проголосовали 2668 пользователей.
Воздержался 721 пользователь.

Теги: [edisonsoftware](#)

Хабы: [Блог компании Edison](#),
[Программирование](#), [Веб-разработка](#),
[Разработка мобильных приложений](#)

Редакторский дайджест ×

Присылаем лучшие статьи раз в месяц

Электропочта

Оставляя свою почту, я принимаю [Политику конфиденциальности](#) и даю согласие на получение рассылок



Edison

Изобретаем успех: софт и стартапы

[Facebook](#) [X](#) [ВКонтакте](#) [Google+](#)



21

Карма

0

Общий
рейтинг

[Подписаться](#)

Thomas Alva
[@ThomasAlva](#)



Пользователь

 Комментарии 48

Публикации

[ЛУЧШИЕ ЗА СУТКИ](#) [ПОХОЖИЕ](#)

OldfagGamer 8 часов назад

Ностальгические игры: Fallout New Vegas



Простой



19 мин



1.9K

[Обзор](#)

+21



4



18 +18



Catx2 2 часа назад

Нижегородское метро: 40 горьких лет



28 мин



1.2K



+18



2



5 +5



Bright_Translate 7 часов назад

Digital Ocean преследует меня из-за \$0,01 или Полезный урок по автоматизации

 Простой 7 мин 3К

Кейс

Перевод



+18



5



5

+5



teamfighter 4 часа назад

Как за 5 дней с помощью Claude я создал приложение для кошки с диабетом (и кажется запустил стартап)

 Средний 6 мин 1К

Кейс



+16



5



7

+7



DRoman0v 8 часов назад

Hi-tech наушники Sony WH-1000XM6: мой опыт использования и впечатления из первых рук

 7 мин 5К

+16



6



27

+27



aak204 9 часов назад

Современные OCR для сложных документов: сравниваем 6 open-source моделей на реальном кошмаре инженера

Средний

6 мин

3.3К

Из песочницы

+16

44

22 +22



samako 23 часа назад

Собираем простейшую RAG-систему на PHP с фреймворком Neuron AI за вечер

Средний

15 мин

3.3К

Туториал

+16

53

4 +4



valisak 3 часа назад

Теория неоднородной Вселенной как альтернатива тёмной энергии

Простой

8 мин

480

Обзор

+14

3

3 +3



zarazaexe 3 часа назад

Что скрывается за «сертификатами безопасности» от Минцифры?

Средний

6 мин

2.2K

Из песочницы



+13



15



12 +12



vsradkevich 23 часа назад

Стандартная модель. От симметрий к кваркам

Сложный

36 мин

1.8K

Обзор



+13



26



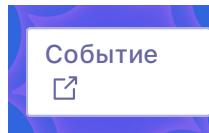
8 +8

Для тех, кто в тяжелой цифре и тех, кто про нее думает: опрос о выборе карьеры

Опрос

Показать еще

МИНУТОЧКУ ВНИМАНИЯ



Качайте
карьеру:
полезные
знакомства,
истории и
лайфхаки

Расскажите,
что вам не
нравится на
собеседованиях

Ты можешь
забрать
хорошие
скидки, но до
сих пор не
забрал?

Ваш аккаунт

[Войти](#)[Регистрация](#)

Разделы

[Статьи](#)[Новости](#)[Хабы](#)[Компании](#)[Авторы](#)[Песочница](#)

Информация

[Устройство сайта](#)[Для авторов](#)[Для компаний](#)[Документы](#)[Соглашение](#)[Конфиденциальность](#)

Услуги

[Корпоративный блог](#)[Медийная реклама](#)[Нативные проекты](#)[Образовательные
программы](#)[Стартапам](#)