



Жизненный цикл программного обеспечения

Выполнила студентка
Гафарова Самира 20П-3

Содержание



Понятие жизненного цикла



Основные этапы



Модели Жизненного цикла



Доп. модели



Конец

Понятие жизненного цикла

Основным понятием программной инженерии является понятие жизненного цикла ПО.

Жизненный цикл ПО (software lifecycle) – это период времени, который начинается с момента принятия решения о необходимости создания ПО и заканчивается в момент его полного изъятия из эксплуатации.

ЖИЗНЕННЫЙ ЦИКЛ ПО



Основные этапы

Обычно к этапам жизненного цикла относят:

1. Анализ требований.
2. Проектирование.
3. Программирование.
4. Тестирование и отладку.
5. Эксплуатацию, сопровождение и поддержку.



Модели жизненного цикла ПО

1. Waterfall (каскадная модель)
2. Итерационная модель
3. спиральная модель
4. инкрементной модели.



Waterfall (каскадная модель)

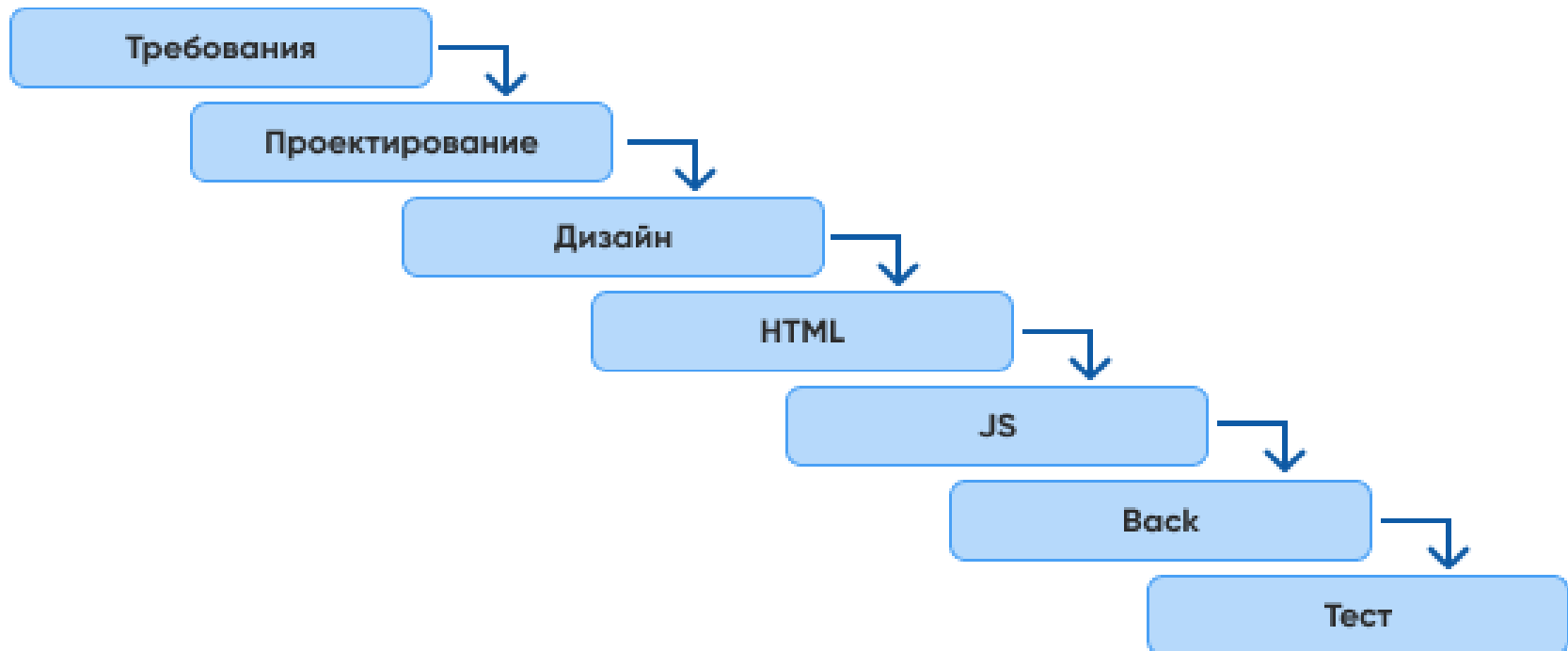
Основная суть модели Waterfall в том, что этапы зависят друг от друга и следующий начинается, когда закончен предыдущий, образуя таким образом поступательное (каскадное) движение вперед.

Параллелизм этапов в каскадной модели, хоть и ограничен, но возможен для абсолютно независимых между собой работ. При этом интеграция параллельных кусков все равно происходит на каком-то следующем этапе, а не в рамках одного.

Недостатками этой модели являются получение результата по прохождению всех этапов и сложность выявления ошибок. Возвращаться назад трудно. Не понятно что возвращать: если произошел сбой на каком-то этапе, его последствия видны только в конце.

Каскадная модель

Как работает изнутри



Как видит клиент

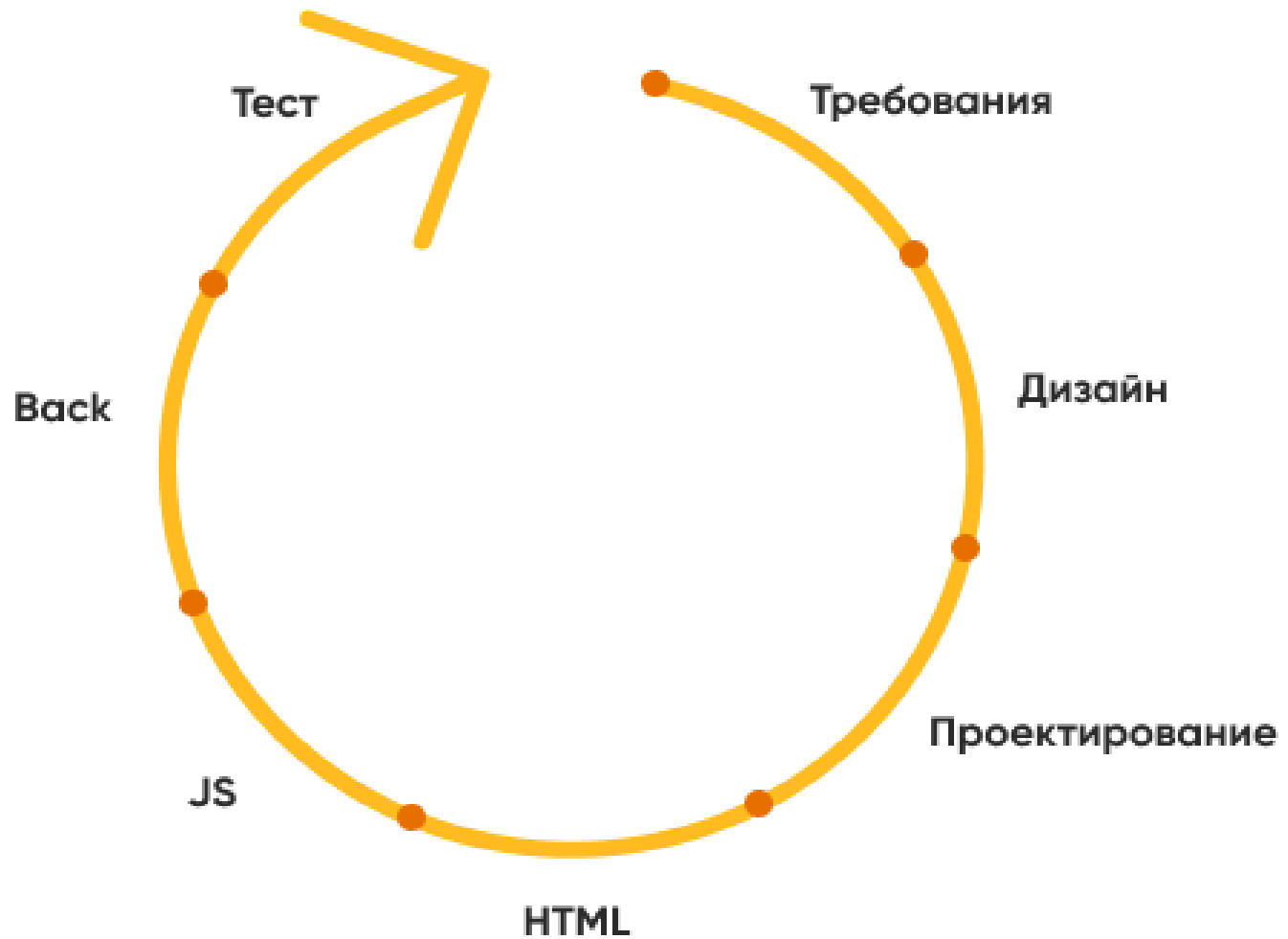


Итерационная модель

Итерационная модель предполагает разбиение проекта на части (этапы, итерации) и прохождение этапов жизненного цикла на каждом из них. Каждый этап является законченным сам по себе, совокупность этапов формирует конечный результат.

Использование итерационной модели снижает риски глобального провала и растраты всего бюджета, получение несинхронизированных ожиданий и ошибочного понимания процессов как клиентом, так и каждым участником команды разработки. Оно также дает возможность завершения разработки в конце любой итерации (в каскадной модели вы должны прежде завершить все этапы).

Итерационная модель

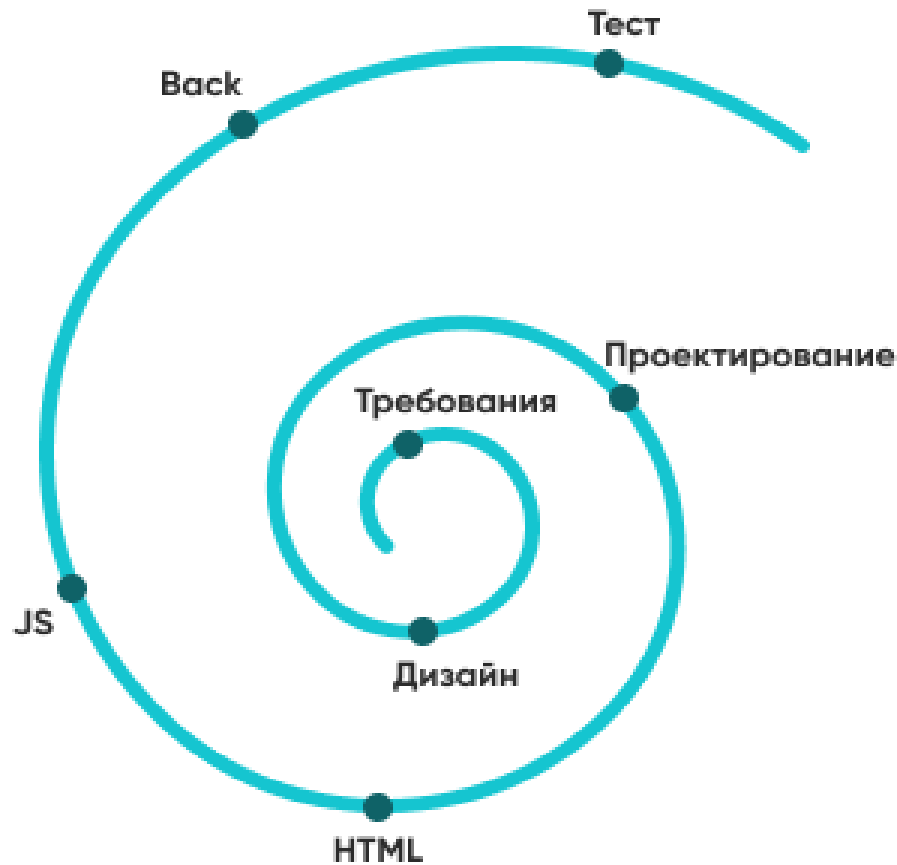


Спиральная модель

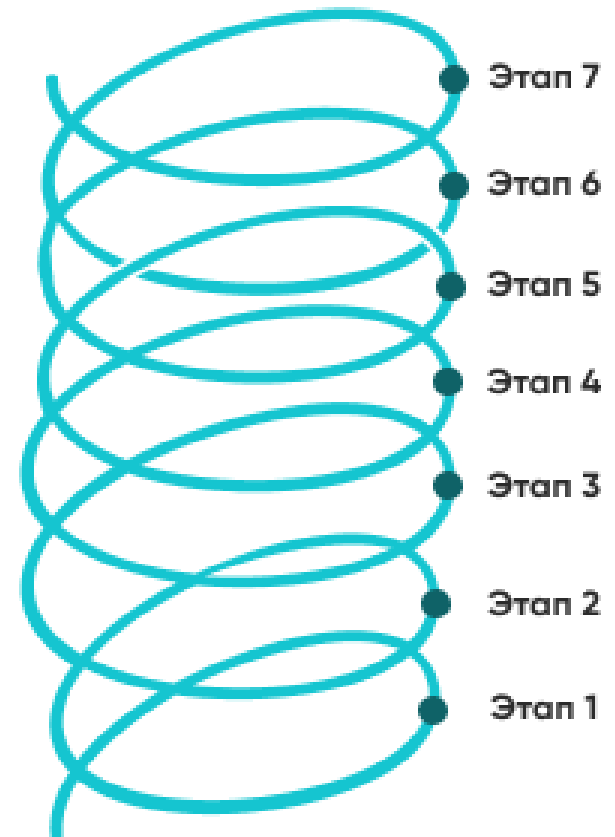
Все этапы жизненного цикла при спиральной модели идут витками, на каждом из которых происходят проектирование, кодирование, дизайн, тестирование и т. д. Такой процесс отображает суть названия: поднимаясь, проходится один виток (цикл) спирали для достижения конечного результата. Причем не обязательно, что один и тот же набор процессов будет повторяться от витка к витку. Но результаты каждого из витков ведут к главной цели.



Спиральная модель



Как работает изнутри



Как видит клиент

Инкрементная модель

Принцип, который лежит в основе инкрементной модели, подразумевает расширение возможностей, доработку модулей и функций приложения. Буквальный перевод слова инкремент: «увеличение на один». Это «увеличение на один» применяется в том числе для обозначения версий продукта.

Если в каскадной модели по сути есть два состояния продукта: «ничего» и «готовый продукт», то с появлением итерационных моделей стало применяться версионирование продукта.

Каждая итерация обозначается цифрой: 1, 2, 3 и соответственно продукт после каждой итерации имеет версию с соответствующим номером: v.1, v.2, v.3. Числами после слова версия обозначают масштабные изменения в ядро продукта.

Дополнительные модели жизненного цикла

Модель Большого Взрыва Модель Большого Взрыва не имеет никаких руководящих принципов вообще. Эта модель жизненного цикла разработки программного обеспечения была задумана для того, чтобы помочь ориентироваться в проектах, где клиент не знает, как будет выглядеть конечный программный продукт.

Гибкая Модель

Гибкая методология разработки — серия подходов к разработке программного обеспечения, ориентированных на использование интерактивной разработки, динамическое формирование требований и обеспечение их реализации в результате постоянного взаимодействия внутри самоорганизующихся рабочих групп, состоящих из специалистов различного профиля.





Благодарю за внимание!