Методология разработки ПО – это структура,согласно которой построена разработка.

**Подход Waterfall**

Основной принцип водопадного подхода — это строгая последовательность этапов разработки, выполняемых в соответствии с согласованным планом. План — это первое, что нужно согласовать. Создание программного обеспечения включает следующие этапы:

* обсуждение идеи создания концепта;
* анализ и планирование требований;
* дизайн;
* кодирование и реализация;
* тестирование;
* выпуск продукта;
* поддержка и сопровождение.

Преимущества:

1. Легко работать,управлять и контролировать
2. Точная документация
3. Результат известен
4. Легко соблюдаемые сроки

Недостатки:

1. Нет права на ошибку
2. Первоначальная информация не всегда точна
3. Клиент не видит работающего ПО допоздна
4. Отсутствие связи
5. Заключительное тестирование

**Подход Agile**

Все начинается с простого дизайна, который будет многократно обновляться по мере развития проекта. Объем работ разделен на модули и выполняется спринтами. Каждый спринт длится пару недель и состоит из следующих этапов:

* планирование — разделить концепцию на маленькие части
* анализ требований — многочисленные встречи с клиентами и менеджерами проектов для сбора подробной информации
* проектирование — создать дизайн по последним требованиям.
* кодирование — для создания рабочего продукта
* тестирование — для проверки готового программного обеспечения на наличие ошибок.
* запуск — доставка готового продукта покупателям

Преимущества:

1. Изменения приветствуются
2. Сильная командная работа
3. Участие клиента
4. Более высокое качество

Недостатки:

1. Отсутствие планирования
2. Заброшенная документация
3. Полная самоотдача
4. Результат может отличаться от ожидания

**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 не идеальна, главное, чтобы она работала. Понимая конечную цель, мы стремимся к ней так, чтобы каждый шаг был результативен, а каждая версия — работоспособна. Используется в следующих схемах:

* Характеристики всего проекта могут быть определены и поняты.
* В первую очередь определены важные шаги, в то время как некоторые необходимые улучшения возникают в процессе разработки.
* Разработчики собираются интегрировать некоторые новые технологии в процессе разработки.
* План может быть изменен в будущем при условии, что не существует большого риска.

Преимущества:

1. В жизненном цикле разработки программного обеспечения можно заранее создать несколько возможностей.
2. Эффективен для постоянно меняющихся требований проекта.
3. Может быть организовано совместное развитие.
4. Опасности распознаются и исправляются путем итерации, и каждая итерация может быть с легкостью запущена.
5. В модели итерации на обдумывание уходит меньшее время, чем на разработку.

Недостатки:

1. Неприменима в небольших проектах
2. Для эстимации проекта требуются опытные кадры
3. Возможно переоценивание некоторых ресурсов для проекта в процессе разработки
4. Может потребоваться большее количество ресурсов, чем было запланировано.