**Модель кодирования и устранения ошибок:**

Эта модель разработки, очень простая. Получаем требования, пишем код, показываем заказчику, исправляем ошибки.

Основным ее плюсом является то, что она не требует затрат на организацию, очень простая и быстрая.

Минус в том, что эта модель неудобна, если проект более-менее большой.

#### Каскадная модель жизненного цикла программного обеспечения (водопад):

#### В этой модели разработка идет одно за другим каскадом. Переход с одного этапа разработки на другой происходит только после полного и успешного завершения предыдущего этапа.

#### Плюсы то что весь процесс разработки представлен в виде четких этапов, требования к проекту четко определены на начальном этапе, что предотвращает изменения в процессе разработки ну и простота управления проектом.

#### Минусы эта модель может оказаться неэффективной при необходимости быстрого реагирования на изменения в требованиях после начала разработки, клиент может видеть результаты только после завершения проекта.

#### Инкрементная модель

#### Это метод, в котором проект проектируется, реализуется и тестируется небольшими добавлениями до самого окончания разработки.

#### Плюсы получение функционального продукта после реализации каждого инкремента, упрощение тестирования инкрементов по сравнению с продуктами промежуточных уровней при разработке систем по методу нисходящего проектирования.

#### Минусы недостаточно чёткое определение требований, отсутствие снижения общих затрат на выполнение проекта.

#### Итеративная модель:

#### это подход к разработке программного обеспечения, когда небольшими циклами совершенствуются рабочие версии, все более и более продвигаясь к идеальному состоянию.

#### Плюсы быстрый выпуск продукта, хотя и с минимальными функциями, более гибкая подстройка продукта под новые запросы пользователей.

#### Минусы отсутствие понятных сразу бюджета и сроков разработки, нередко приходится в значительной мере переписывать решения, уже сделанные в предыдущих итерациях.

#### V-Модель:

#### Левая часть модели как бы дублирует каскадную, а правая показывает, какие задачи стоят перед командой тестирования на каждом этапе

#### V-модель сохранила все плюсы каскадной модели. А еще улучшилось качество, ведь теперь тестирование происходит сразу на каждом этапе, и не нужно ждать завершения проекта.

#### Минусы этой модели такие же, как и у каскадной.

#### Спиральная модель жизненного цикла программного обеспечения:

#### Спиральная модель это комбинация водопадной модели и итеративной модели. Спиральная модель помогает адаптировать элементы разработки программного обеспечения из нескольких моделей процессов для программного проекта на основе уникальных шаблонов рисков, обеспечивая эффективный процесс разработки.

#### Плюсы заказчик может увидеть работающую версию продукта уже на ранних стадиях жизненного цикла ПО, изменения могут быть внесены на поздних стадиях разработки.

#### Минусы на самых ранних стадиях дата завершения работы над проектом может быть неизвестна, что также усложняет контроль над процессом разработки, мониторинг рисков требует дополнительных ресурсов, а значит, эта модель может оказаться весьма затратной.

#### Модель хаоса:

#### Главное правило — это всегда решать наиболее важную задачу первой.

#### Плюсы гибкая адаптация под любые измениния, работа с рисками.

#### Минусы неопределенность, риск недостаточного планирования.

#### V модель мне симпатизирует больше всех так как на каждом этапе разработки есть возможность протестировать и сразу устранить все недостатки.