



НЕТОЛОГИЯ

# ЖИЗНЕННЫЙ ЦИКЛ РАЗРАБОТКИ ПО. ГИБКИЕ МЕТОДОЛОГИИ РАЗРАБОТКИ ПО



АНАСТАСИЯ ШАРИКОВА



# АНАСТАСИЯ ШАРИКОВА

QA Lead в Bookmate



[shharikova@gmail.com](mailto:shharikova@gmail.com)



[@shharikova](https://www.instagram.com/shharikova)



# ПЛАН ЗАНЯТИЯ

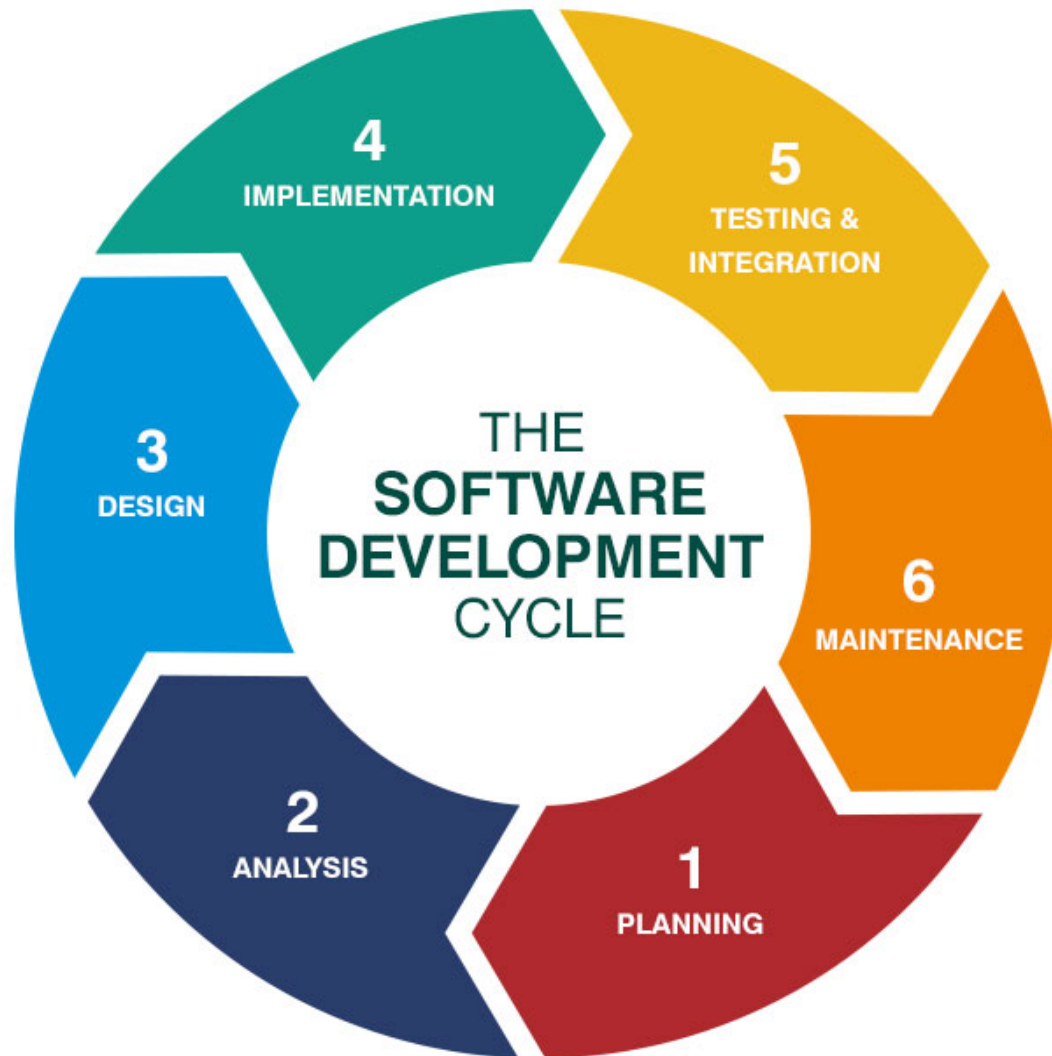
1. SDLC – жизненный цикл разработки ПО
2. Стадии разработки
3. Фазы разработки
4. Методологии разработки ПО
5. STLC – жизненный цикл тестирования ПО
6. Стадии тестирования
7. Жизненный цикл дефекта
8. Как вписать STLC в SDLC и не умереть



# ВСПОМИНАЕМ ПРОШЛЫЕ ЗАНЯТИЯ

- Методы и виды тестирования ПО;
- Основы клиент-серверного взаимодействия;
- Техники тест дизайна;
- Подготовка тестовых сценариев;
- Инструменты тестирования ПО.

# SDLC - ЖИЗНЕННЫЙ ЦИКЛ РАЗРАБОТКИ ПО





# СТАДИИ РАЗРАБОТКИ

- Анализ требований к проекту;
- Проектирование;
- Реализация (дизайн и кодирование);
- Тестирование и интеграция;
- Внедрение и поддержка.



# АНАЛИЗ ТРЕБОВАНИЙ К ПРОЕКТУ

- Анализ требований к проекту;
- Подготовка к дальнейшей работе;
- Документирование и анализ;
- Выявление и разрешение противоречий.



# ПРОЕКТИРОВАНИЕ

- Выбор технологий;
- Загрузка команды, бюджетное планирование;
- Разработка проектного решения.





# РЕАЛИЗАЦИЯ

- Разработка back-end;
- Разработка front-end;
- Unit-тесты;
- Развертка кода в программной среде.



# ТЕСТИРОВАНИЕ И ИНТЕГРАЦИЯ

- Поиск и документирование дефектов;
- Ретест дефектов после исправлений;
- Написание/актуализация тестов.



## ВНЕДРЕНИЕ И ПОДДЕРЖКА

- Релиз для конечных пользователей;
- Работа саппорта.

# ФАЗЫ РАЗРАБОТКИ

— **Последовательные фазы:**

1. сначала собираем и разрабатываем требования,
2. потом проектируем модели,
3. затем реализуем их,
4. перед передачей заказчику тестируем и исправляем ошибки.

— **Параллельные фазы:** пока разработчики реализуют требования к текущей версии, аналитики собирают и разрабатывают требования к новой версии. Одновременно над продуктом может работать несколько команд, каждая из которых разрабатывает свою фичу от начала и до конца.

— **Смешанные фазы:** участники команды могут не выделять отдельные фазы, а за счет коммуникационных практик выполнить сразу сбор, разработку требований и проектирование будущей функциональности.

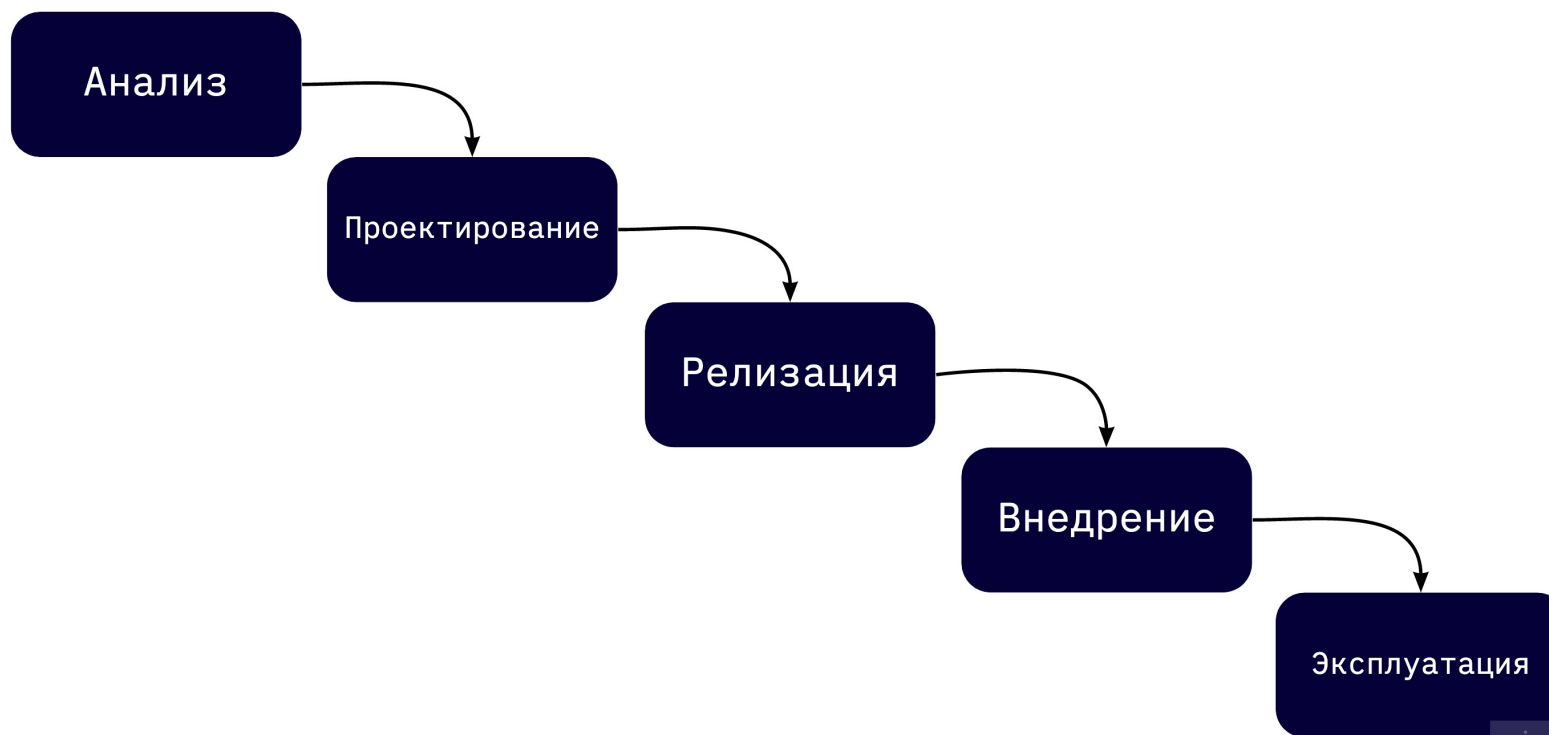
Фазы разработки могут охватывать как целый продукт (водопадная модель), так и минимальную полезную функциональность (Scrum).



# МЕТОДОЛОГИИ РАЗРАБОТКИ ПО

- Waterfall;
- V-model;
- Спиральная модель.
- Инкрементная модель;
- Agile;

# WATERFALL



# V-MODEL



# СПИРАЛЬНАЯ МОДЕЛЬ



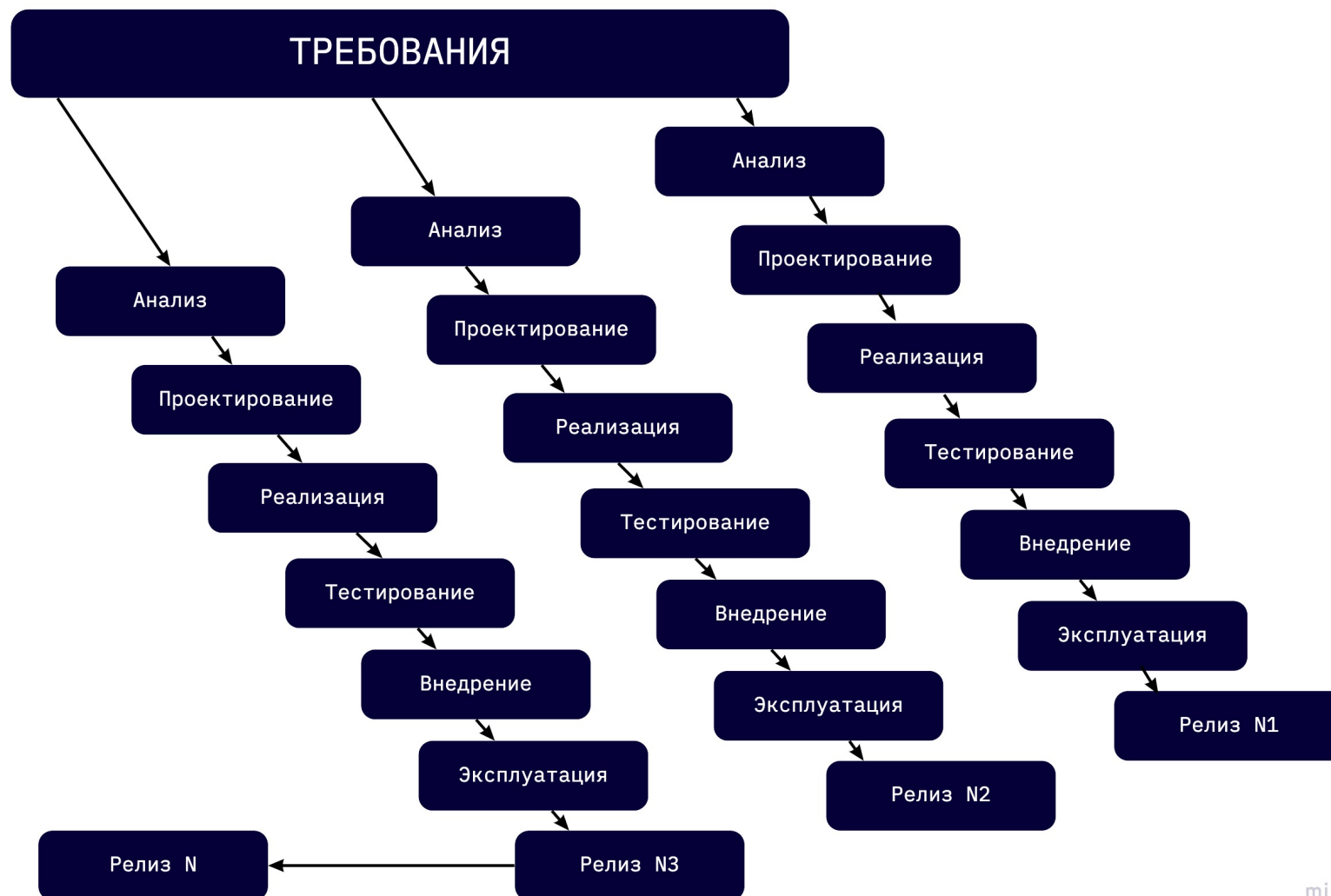




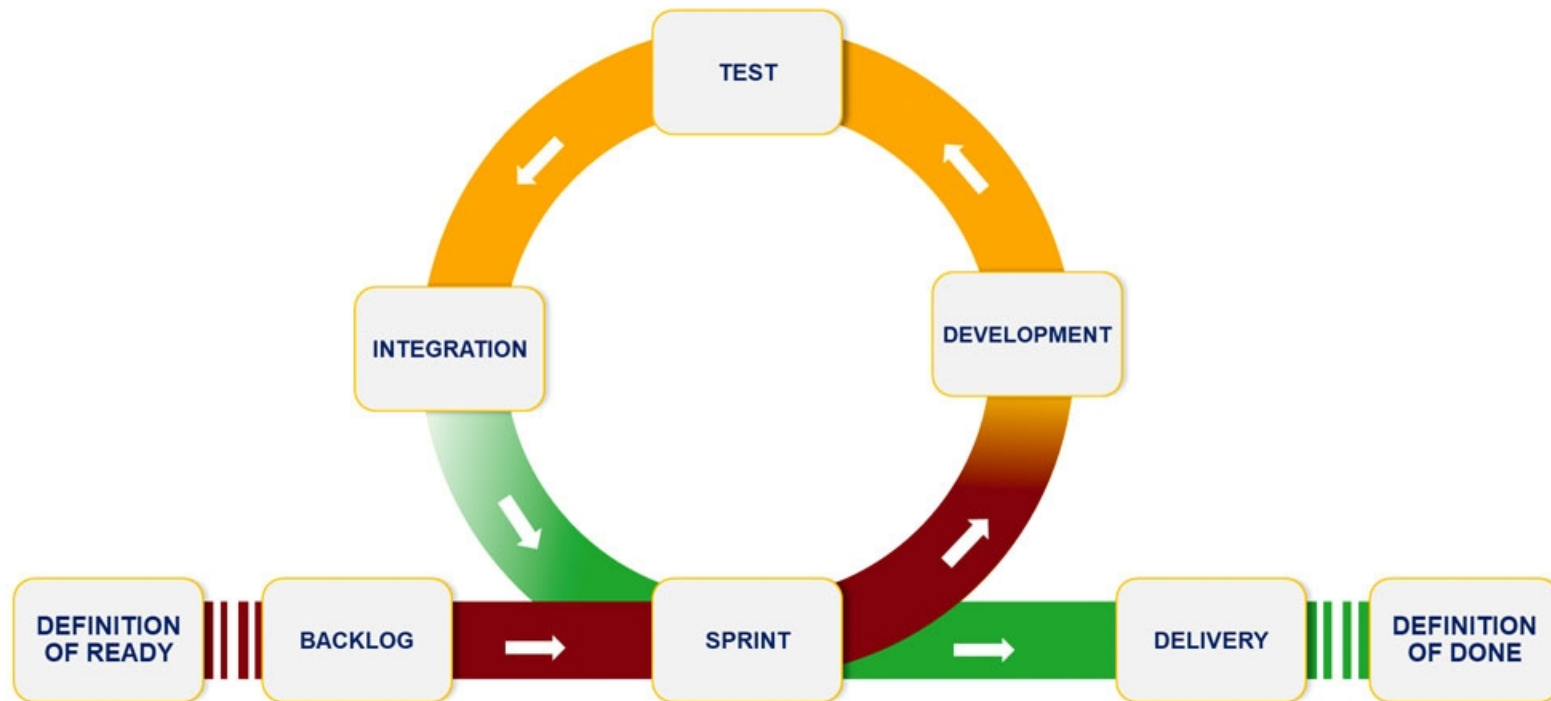
# СПИРАЛЬНАЯ МОДЕЛЬ

1. Сбор требований и планирование;
2. Планирование на основе изменений
3. Анализ риска на основе начальных требований;
4. Анализ риска на основе изменений;
5. Переход к готовой система;
6. -8. Версии системы;
7. Оценка заказчиком.

# ИНКРЕМЕНТНАЯ МОДЕЛЬ



# AGILE



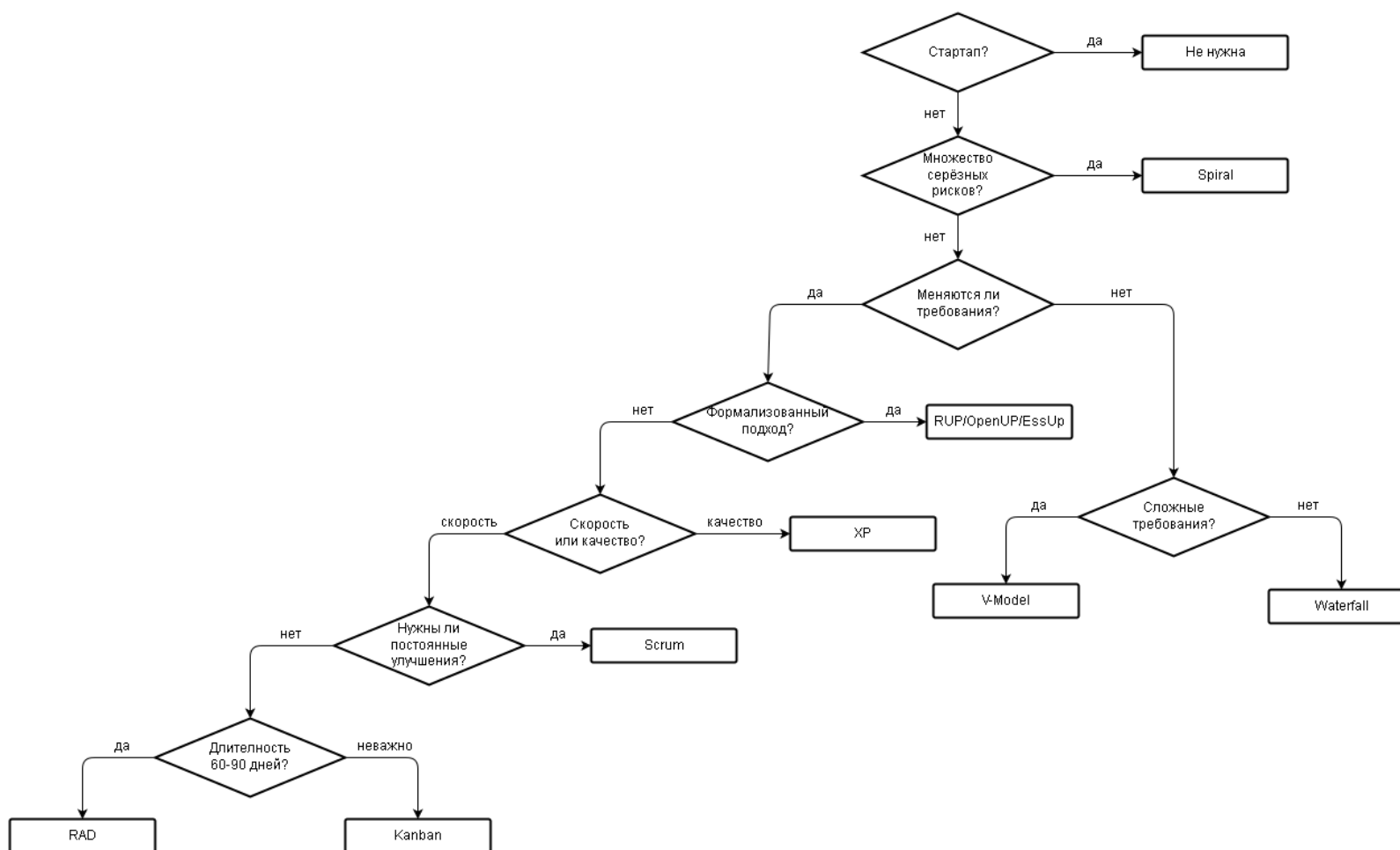


# AGILE

Популярные Agile-методы:

- XP – Экстремальное программирование;
- DSDM – Dynamic Systems Development Method, методика разработки программного обеспечения, основанная на концепции быстрой разработки приложений (RAD);
- Scrum;
- FDD – Feature driven development;
- TDD – Test-driven development;
- BDD – Behavior-driven development;
- Lean software development.

# ОТ ЧЕГО ЗАВИСИТ ВЫБОР МОДЕЛИ РАЗРАБОТКИ



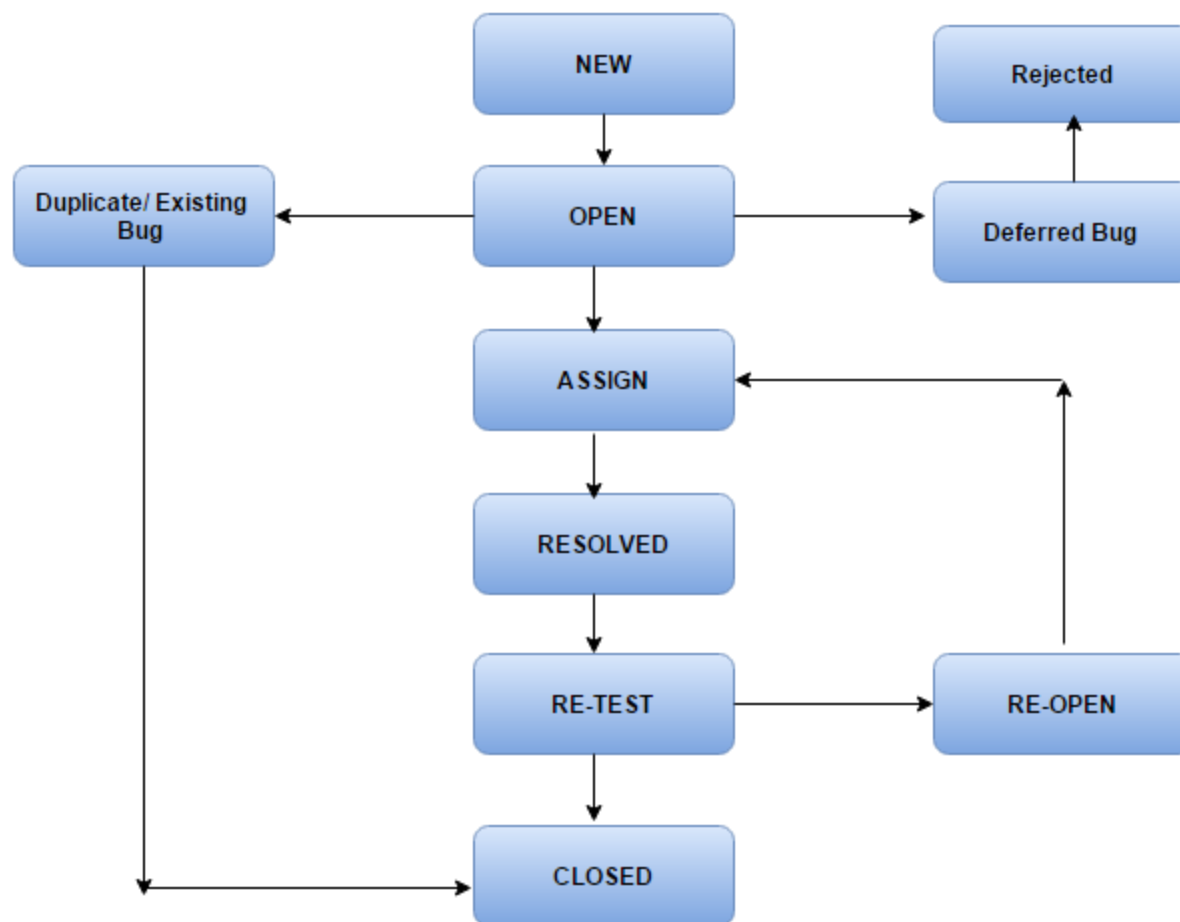
# STLC – ЖИЗНЕННЫЙ ЦИКЛ ТЕСТИРОВАНИЯ ПО



# СТАДИИ ТЕСТИРОВАНИЯ

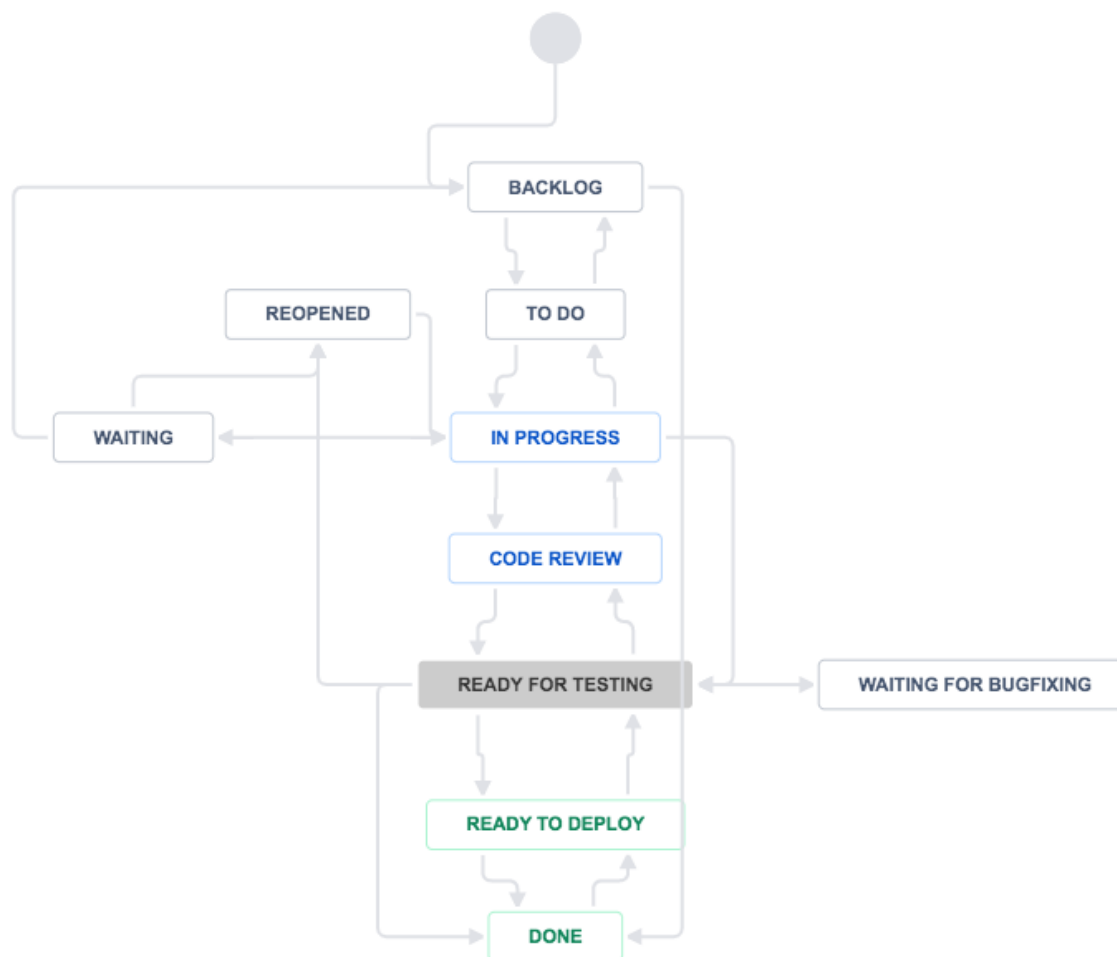
- **Работа с требованиями:** тестирование на соответствие бизнес-целям, полноту охвата, уместность использования, целостность и непротиворечивость.
- **Планирование тестирования:** определение требований к тестам, оценка рисков, выбор стратегии тестирования, создание расписания тестирования, разработка Плана тестирования.
- **Разработка тестов:** определение и описание тестовых случаев, обзор и оценка тестового покрытия, создание и подготовка наборов данных.
- **Установка тестового окружения:** подготовка тестовой среды, подтверждение правильности сборки.
- **Проведение тестов:** выполнение тестов, их оценка, проверка результатов, запись ошибок.
- **Завершение цикла тестирования:** определение критериев завершения и успешности тестирования.

# ЖИЗНЕННЫЙ ЦИКЛ ДЕФЕКТА





# ЖИЗНЕННЫЙ ЦИКЛ ДЕФЕКТА





## КАК ВПИСАТЬ STLC В SDLC И НЕ УМЕРЕТЬ

Полный цикл тестирования обычно совпадает с итерацией разработки или соответствует ее определенной части.

Подход к проверке работоспособности программного продукта похож на оценку продукта от конечного пользователя, поэтому стоит привлекать специалиста к работе на самом раннем этапе — в ходе сбора и анализа требований.

Идеально, когда обсуждение компонентов системы проходит с участием разработчика, пользователя и QA-аналитика.

# ДОМАШНЕЕ ЗАДАНИЕ

Давайте посмотрим ваше [Домашнее задание](#)

- Вопросы по домашней работе задаем в чате Slack
- Задания можно сдавать по частям
- Также пройдите тестирование в личном кабинете «Нетологии»
- Зачет по домашней работе проставляется после того, как приняты **все задания**



Спасибо за внимание!

Время задавать вопросы

**АНАСТАСИЯ ШАРИКОВА**



[shharikova@gmail.com](mailto:shharikova@gmail.com)



[@shharikova](https://www.instagram.com/shharikova)