

## ЛЕКЦИЯ 32

**Тема:** Компонентно-ориентированная модель. Экстремальное программирование XP.

**Цель:** изучить стратегии проектирования программного продукта

**Компонентно-ориентированная модель** является развитием спиральной модели.

Программные компоненты, созданные в реализованных программных проектах, хранятся в библиотеке. В новом программном проекте, исходя из требований заказчика, выявляются кандидаты в компоненты. Далее проверяется наличие этих кандидатов в библиотеке. Если они найдены, то компоненты извлекаются из библиотеки и используются повторно. В противном случае создаются новые компоненты, они применяются в проекте и включаются в библиотеку.



**Рис. 7. Компонентно-ориентированная модель**

**Достоинства** компонентно-ориентированной модели:

1. уменьшает на 30% время разработки программного продукта;
2. уменьшает стоимость программной разработки до 70%;
3. увеличивает в полтора раза производительность разработки.

## **ЭКСТРЕМАЛЬНОЕ ПРОГРАММИРОВАНИЕ XP.**

**Экстремальное программирование (eXtreme Programming, XP) — облегченный (подвижный) процесс (или методология).**

Данная методология отличается от остальных моделей, что она применяется только для разработки ПО, оно не может использоваться в другом бизнесе, как КанБан или Scrum методологии.

В экстремальном программировании смысл в том, что цели задачи заказчика постоянно меняются и разработчикам нужно менять свои подходы и методы разработки, при этом не теряя качества и скорости разработки.

**Цель методики XP — справиться с постоянно меняющимися требованиями к программному продукту и повысить качество разработки.**

**XP-процесс ориентирован на группы малого и среднего размера, строящие программное обеспечение в условиях неопределенных или быстро изменяющихся требований. XP-группу образуют до 10 сотрудников, которые размещаются в одном помещении.**

**Основная идея XP — устранить высокую стоимость изменения, характерную для приложений с использованием объектов, паттернов\* и реляционных баз данных.**

**XP-группа имеет дело с изменениями требований на всем протяжении итерационного цикла разработки, причем цикл состоит из очень коротких итераций.**

Один виток (цикл) разработки короткий, чтобы можно было максимально подстраиваться под новые требования. Скорость итерации обеспечивается с помощью : кодирования, тестирования, общения с заказчиком и проектирования.

**Четырьмя базовыми действиями в XP-цикле являются:**

1. кодирование,
2. тестирование,
3. общения с заказчика
4. проектирование.

**Динамизм обеспечивается с помощью четырех характеристик: непрерывной связи с заказчиком ( и в пределах группы), простоты (всегда выбирается минимальное решение), быстрой обратной связи ( с помощью модульного и функционального тестирования), смелости в проведении профилактики возможных проблем.**

## структуру «идеального» XP-процесса.

Основная структура заложена на итерации разработки ПО. Это наш цикл повторов определенных действий с достижение каких либо результатов.

А в реализацию входит 3 фазы итерации: исследование, блокировка и регулирование.

Основным структурным элементом процесса является XP-реализация, в которую многократно вкладывается базовый элемент — XP-итерация.

**В состав XP-реализации и XP-итерации входят три фазы :**

1. исследование,
2. блокировка,
3. регулирование.

**Исследование** — это поиск новых требований (историй, задач), которые должна выполнять система.

**Блокировка** — выбор для реализации конкретного подмножества из всех возможных требований (иными словами, планирование).

Какие результаты мы планируем получить, с чем сравнивать, те результаты, которые получаем в процессе разработки. Т.е. идеальные параметры, которые мы хотим достичь.

**Регулирование** — *проведение разработки, воплощение плана в жизнь.*

## Повседневная жизнь XP команды

Витки планирования и обратной связи в экстремальном программировании



**Преимущества экстремального программирования** имеют смысл, когда команда полноценно использует хотя бы одну из практик XP.

1. заказчик получает именно тот продукт, который ему нужен, даже если в начале разработки сам точно не представляет его конечный вид
2. команда быстро вносит изменения в код и добавляет новую функциональность за счет простого дизайна кода, частого планирования и релизов
3. код всегда работает за счет постоянного тестирования и непрерывной интеграции
4. команда легко поддерживает код, т.к. он написан по единому стандарту и постоянно рефакторится
5. быстрый темп разработки за счет парного программирования, отсутствия переработок, присутствия заказчика в команде
6. высокое качество кода
7. снижаются риски, связанные с разработкой, т.к. ответственность за проект распределяется равномерно и уход/приход члена команды не разрушит процесс
8. затраты на разработку ниже, т.к. команда ориентирована на код, а не на документацию и собрания

Несмотря на все плюсы, XP не всегда работает и имеет ряд слабых мест. Итак, экстремальное программирование — **недостатки**:

1. успех проекта зависит от вовлеченности заказчика, которой не так просто добиться
2. трудно предугадать затраты времени на проект, т.к. в начале никто не знает полного списка требований
3. успех XP сильно зависит от уровня программистов, методология работает только с senior специалистами
4. менеджмент негативно относится к парному программированию, не понимая, почему он должен оплачивать двух программистов вместо одного
5. регулярные встречи с программистами дорого обходятся заказчикам
6. из-за недостатка структуры и документации не подходит для крупных проектов

#### **Контрольные вопросы:**

1. что такое рефакторинг?
2. Расскажите про компонентно ориентированную модель
3. Что такое XP модель?
4. В чём суть метода экстремального программирования?
5. Базовые действия XP ?
6. Цель методики XP?
7. Назовите и опишите достоинства и недостатки

#### **Список использованных источников:**

1. Технологии разработки программного обеспечения С.А. Орлов
2. Технологии разработки программного обеспечения В.В. Бахтизин, Л.А. Глухова
3. <https://finswin.com/projects/osnovnye/sroki-realizacii-proekta.html>