

A close-up, slightly high-angle shot of a factory conveyor belt. The belt is blue and curves from the bottom left towards the top right. It is filled with numerous aluminum cans, likely for soda or beer, which are arranged in neat rows. The cans are silver and have a visible pull-tab on the lid. The background is blurred, showing more of the factory environment with greenery visible in the distance. The text "Software Factory" is overlaid in a large, blue, sans-serif font, centered over the middle of the image.

Software Factory

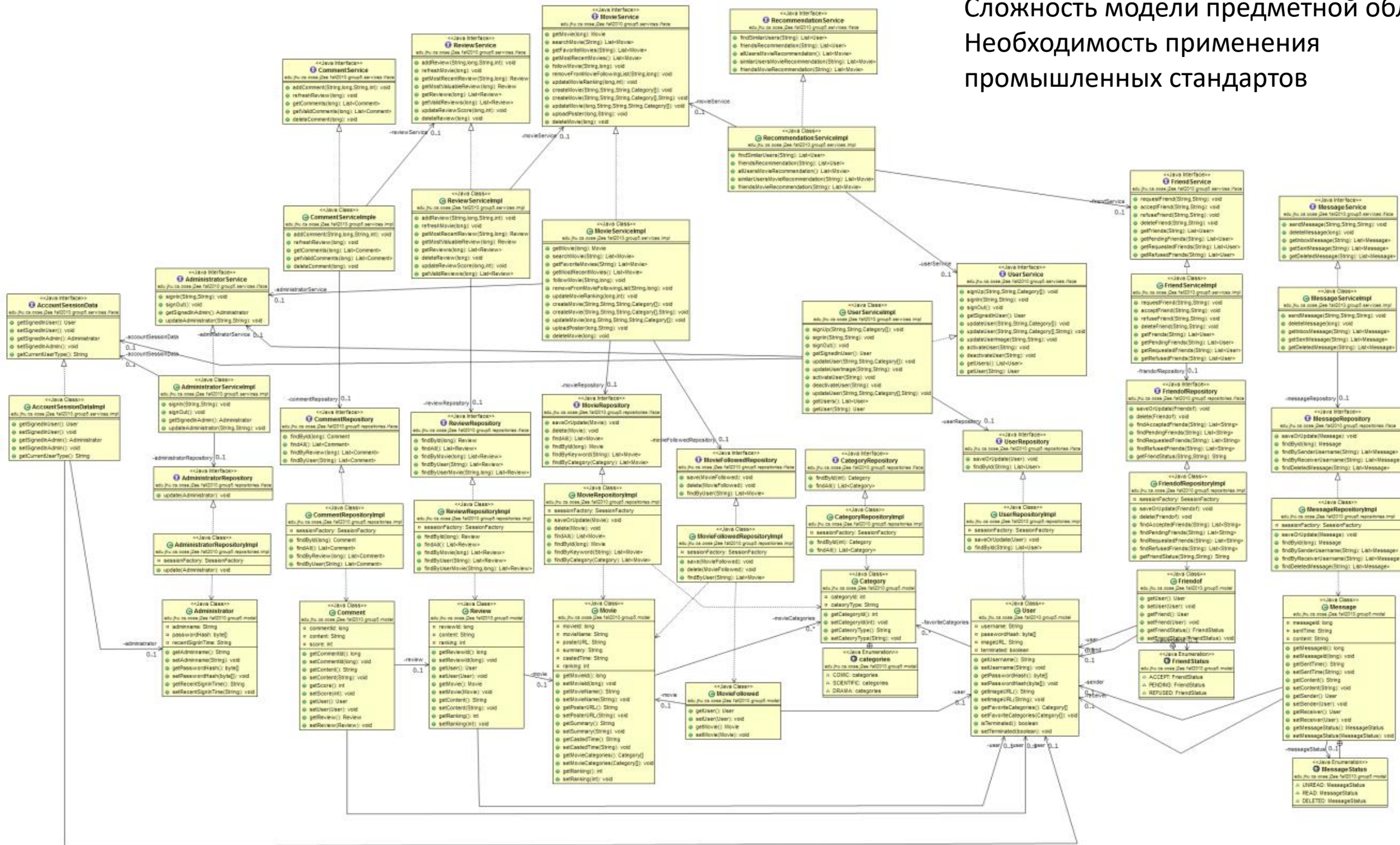
Введение

- Определение
- Контекст и мотивация

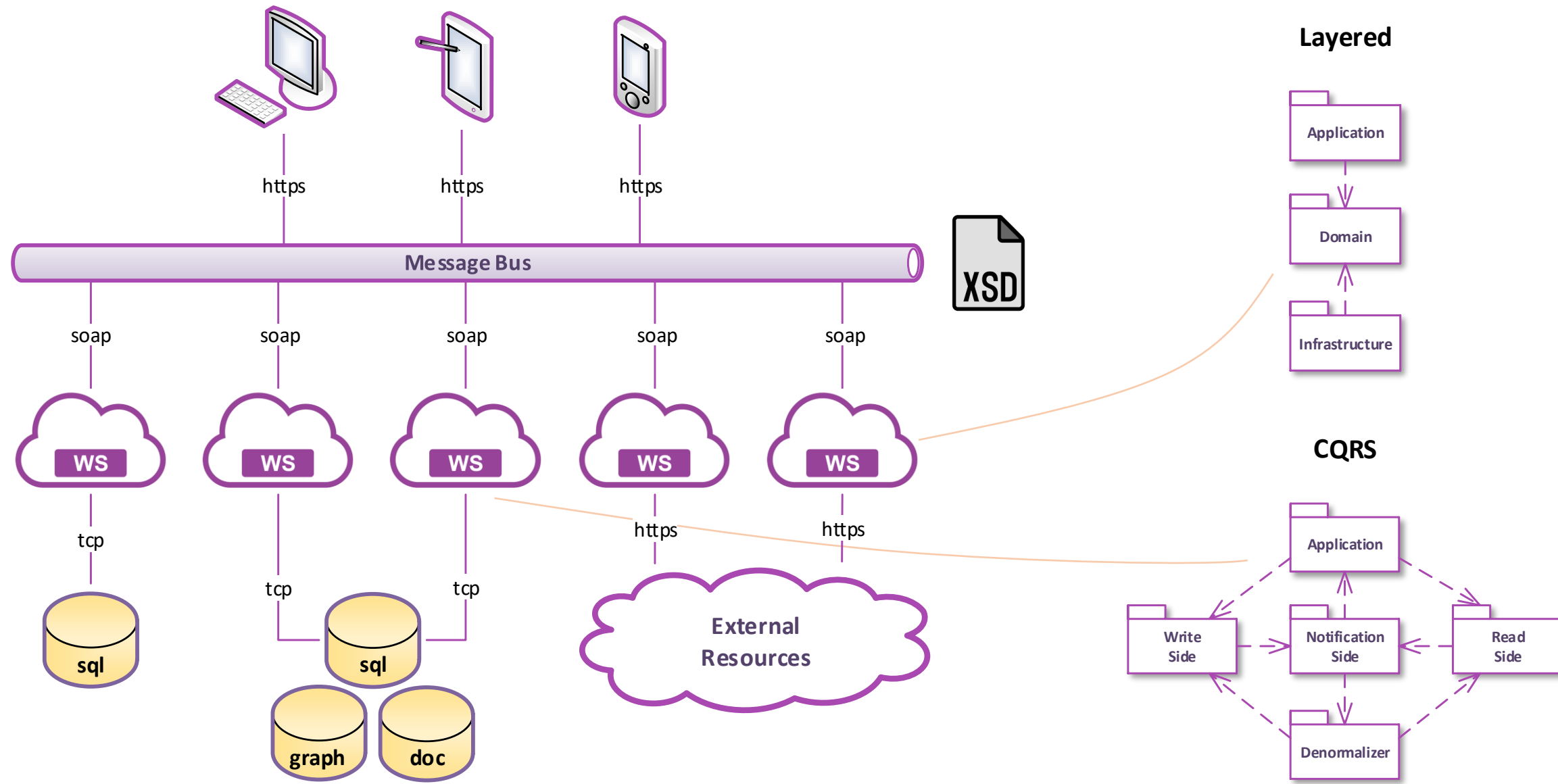
Проблема

- Предыстория
- Особенности проекта
- Проблемы проекта

Сложность модели предметной области
Необходимость применения
промышленных стандартов



Промышленная многопользовательская ERP со сложной доменной моделью и двунаправленными интеграциями



Решение

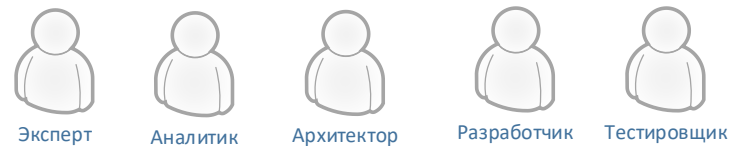
- Общая идея
- План

- Карты процессов
- Референсные имплементации
- Архитектурные шаблоны
- Методические рекомендации
- Автоматические сценарии, скрипты, визарды
- Шаблоны документов, артефактов, процессов
- Визуальные инструменты
- Расширения отладчика
- Повторно-используемый код
- Обучение

Детали

**Управление жизненным циклом
создания программного продукта
(ALM)**

Команда



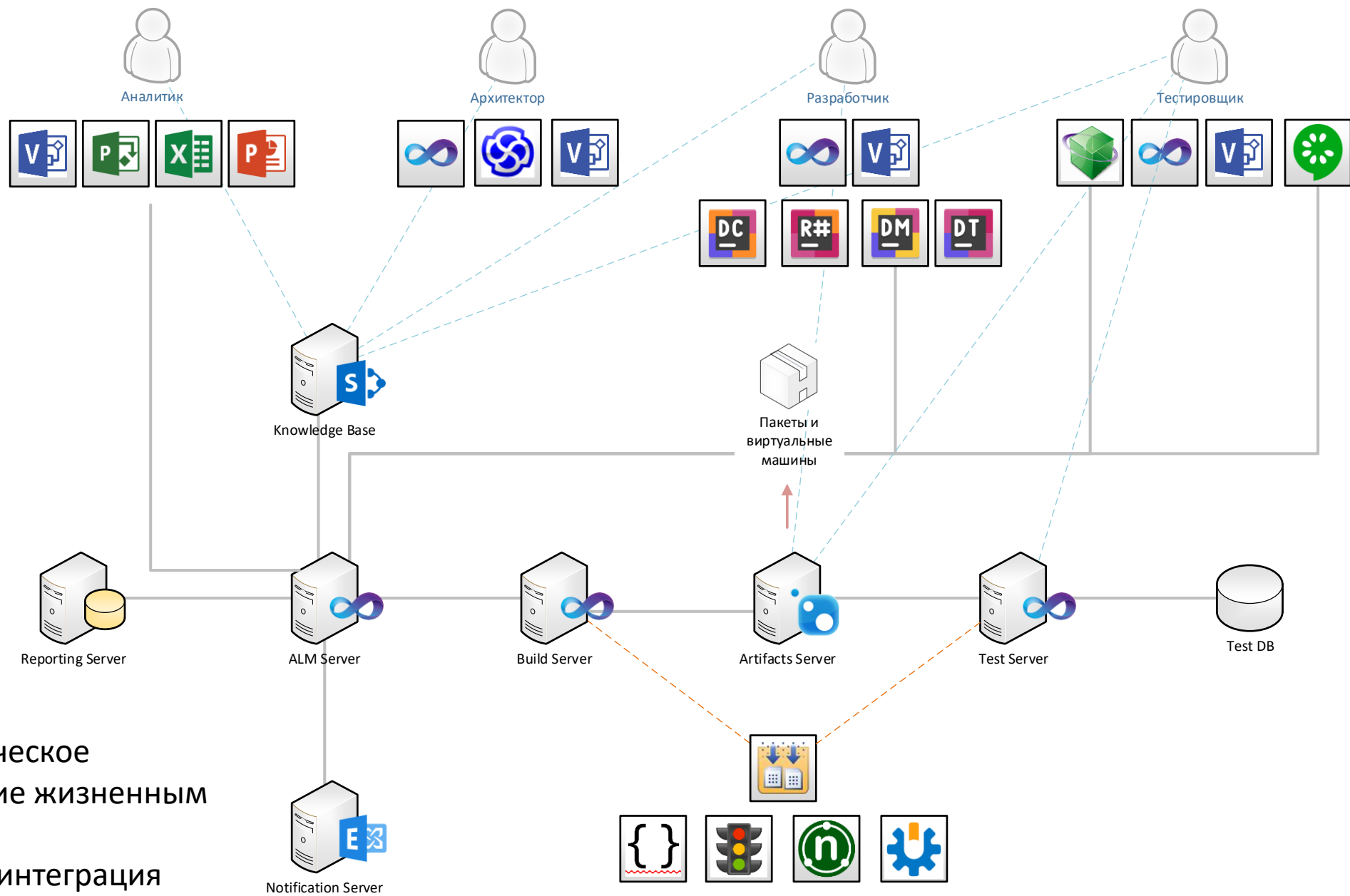
~ 60 человек
4-7 в команде

Процесс

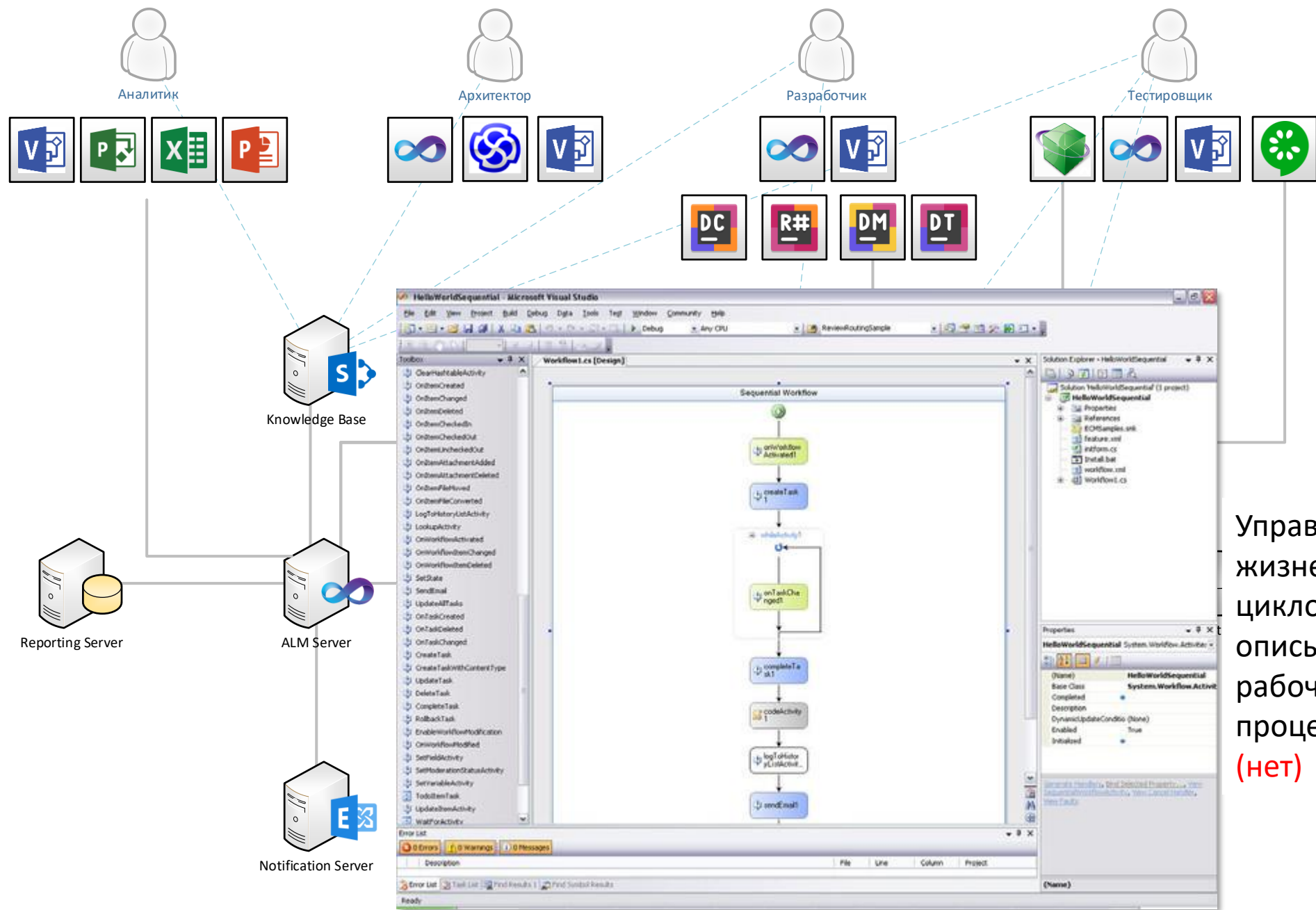


Удобен для трассировки требований

Экосистема разработки

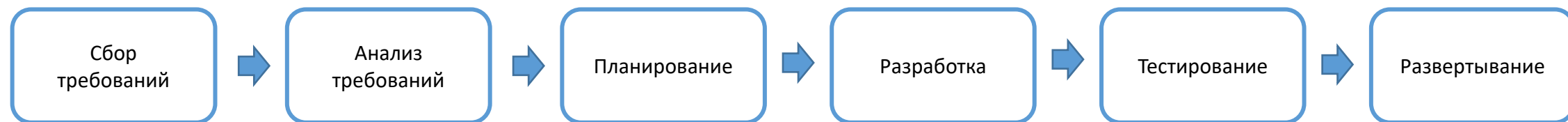


- Автоматическое управление жизненным циклом
- Сквозная интеграция Трассировка

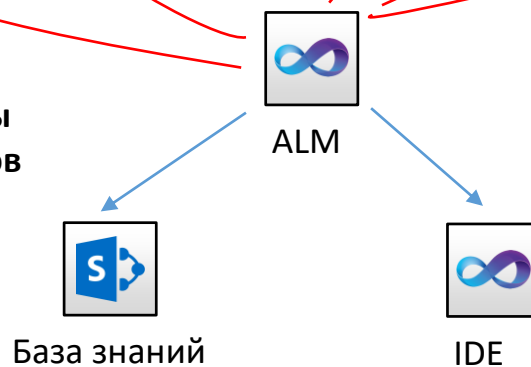


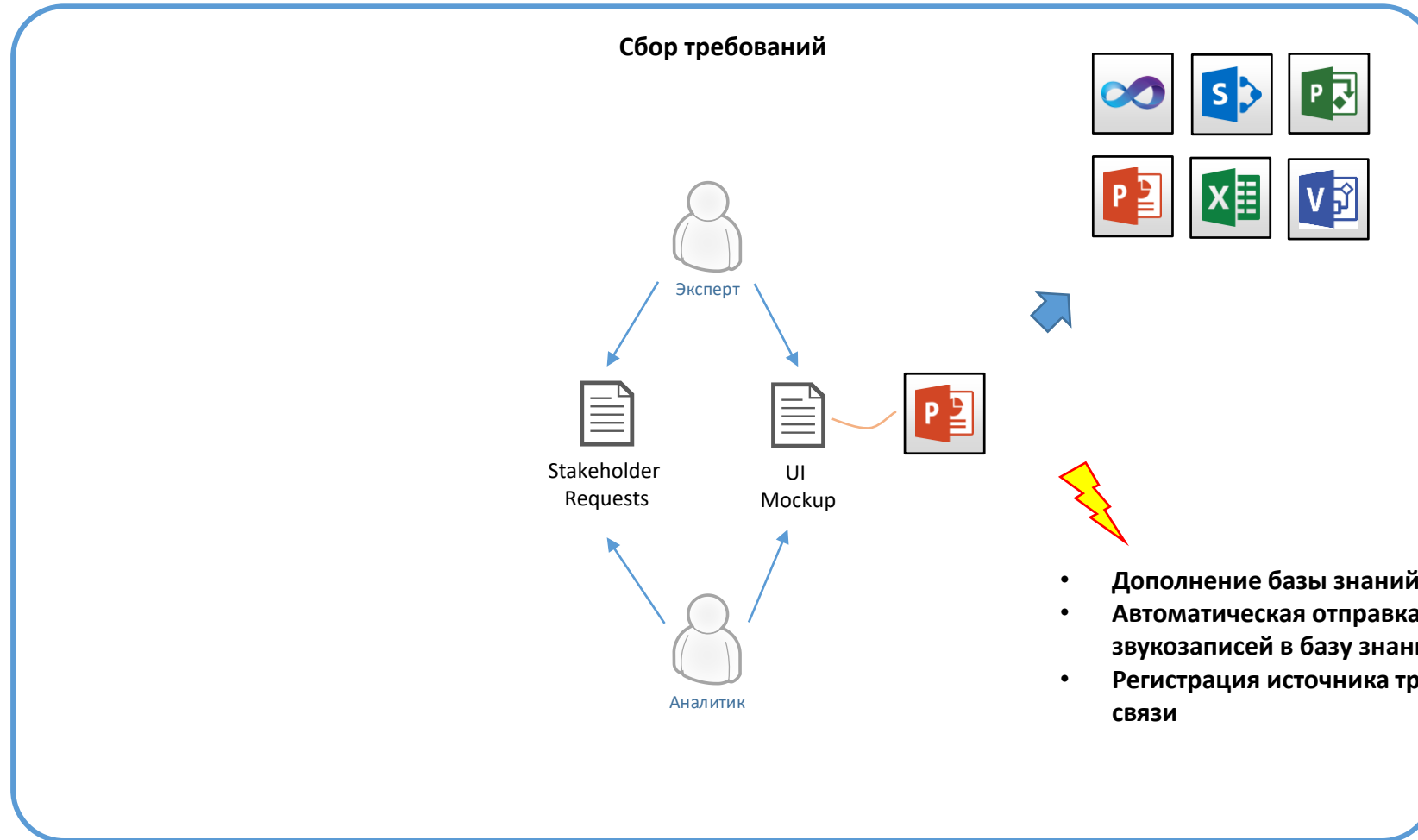
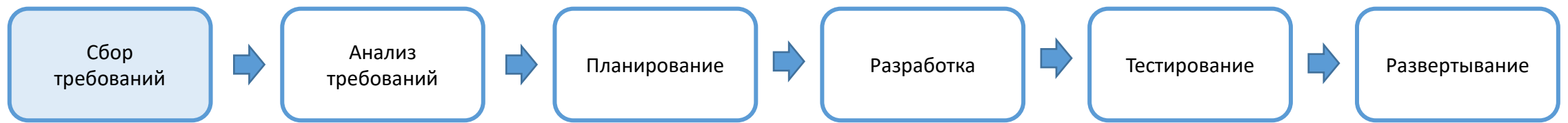
Управление
жизненным
циклом частично
описывается
рабочими
процессами TFS
(нет)

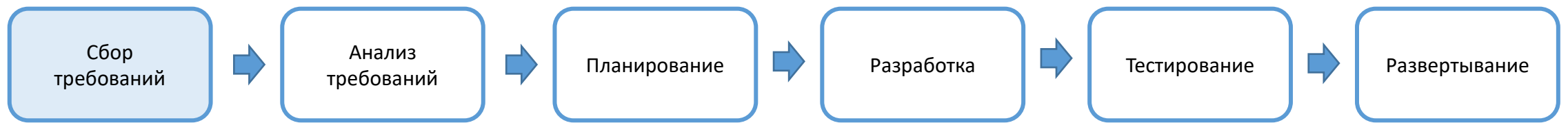
**Автоматизация
каждого этапа
жизненного цикла**



- Карты процессов
- Референсные имплементации
- Архитектурные шаблоны
- Методические рекомендации
- Автоматические сценарии, скрипты, визарды
- Шаблоны документов, артефактов, процессов
- Визуальные инструменты
- Расширения отладчика
- Повторно-используемый код
- Обучение



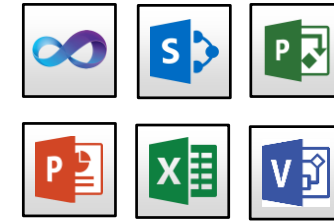
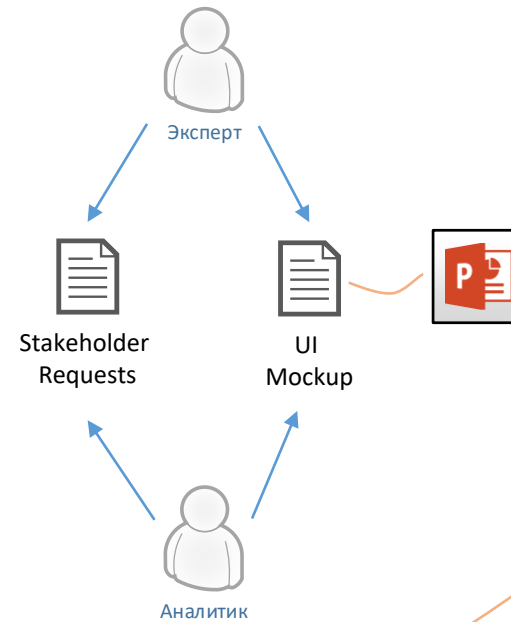




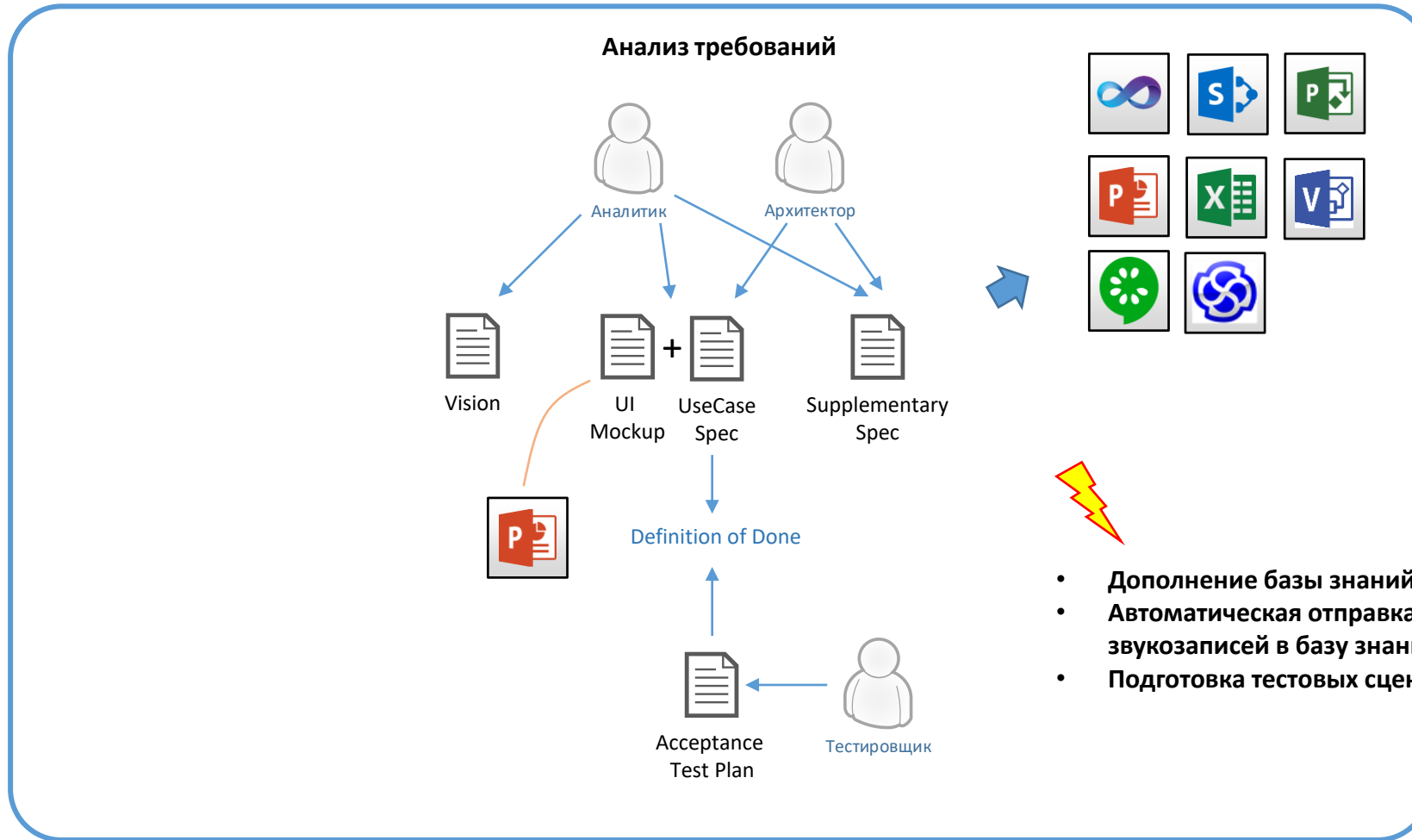
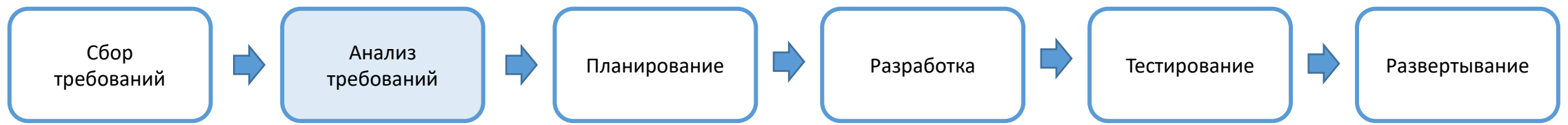
MS InfoPath

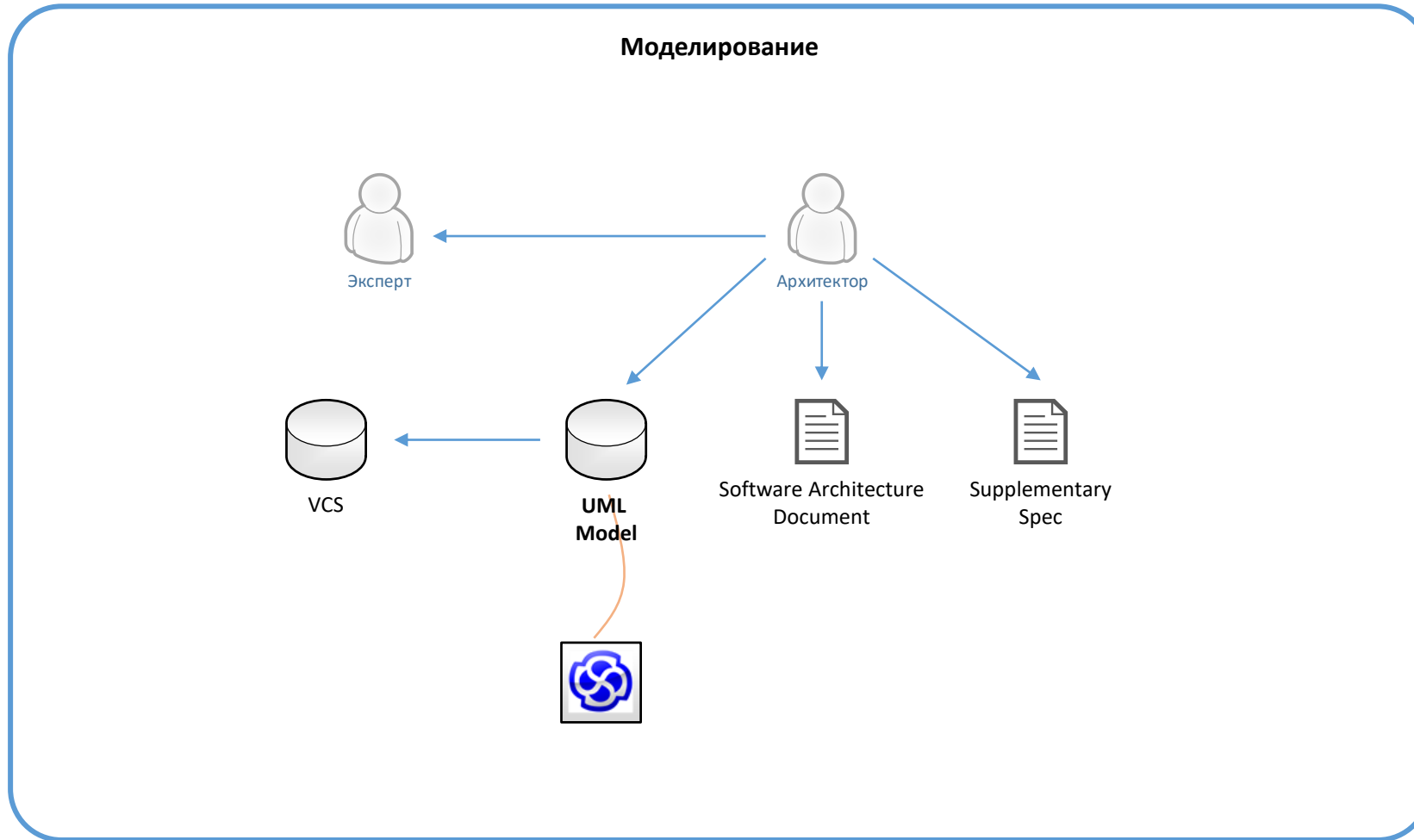
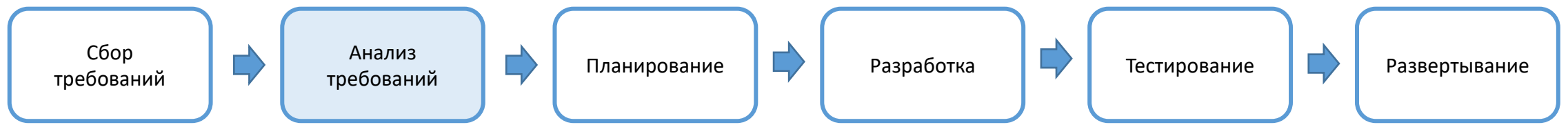
Data	Descrizione	Categoria	Costi
			0,00
Subtotale			
Meno anticipo contanti			0,00
Totale spese			

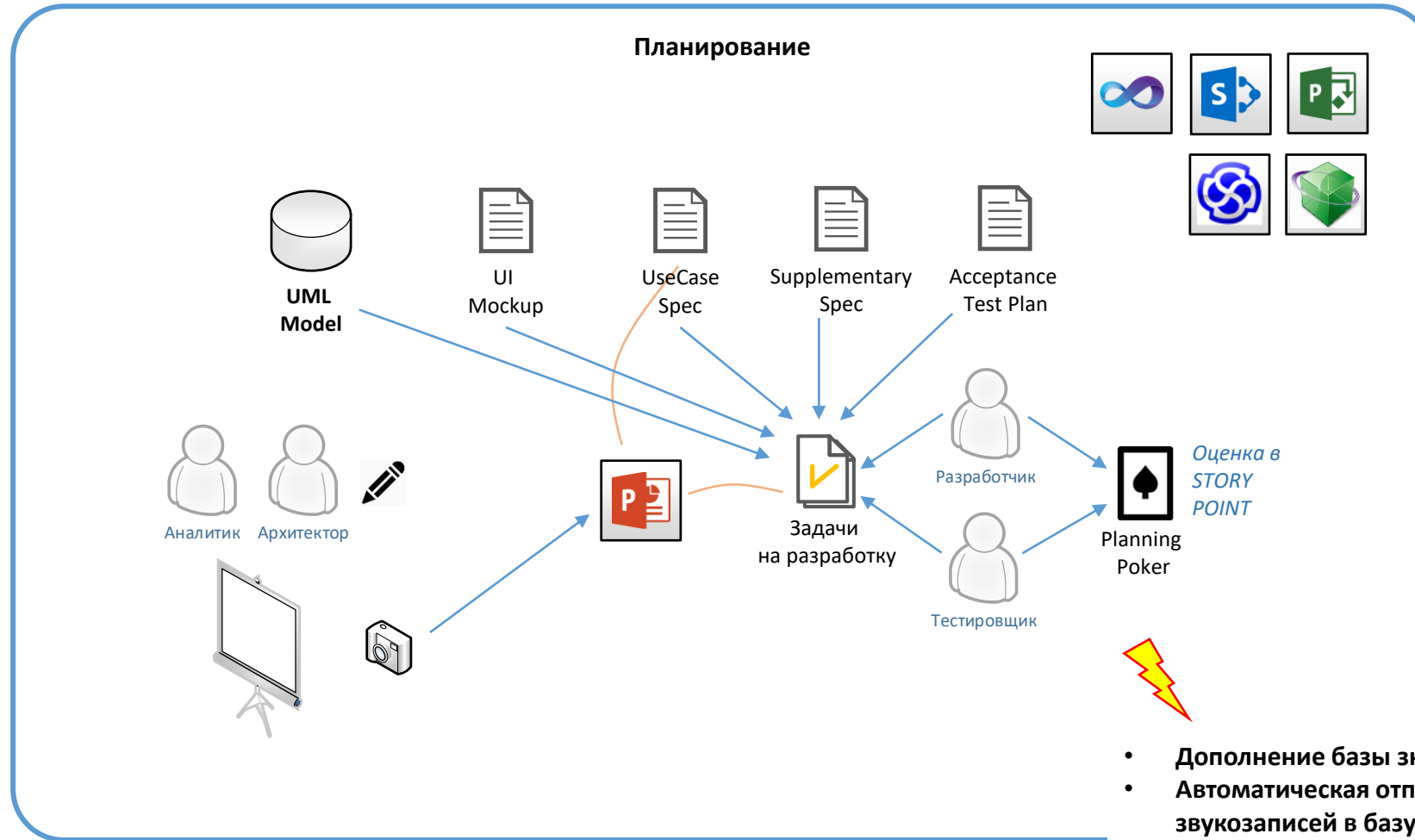
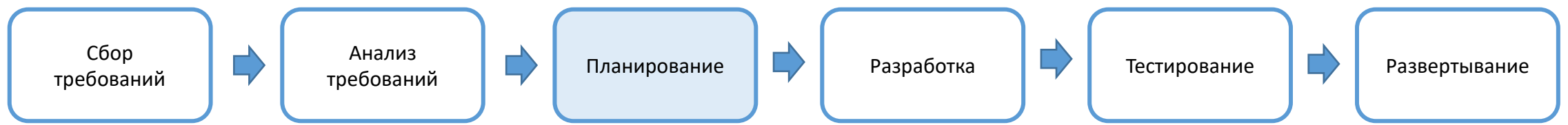
Сбор требований



- Дополнение базы знаний по шаблонам RUP
- Автоматическая отправка фотографий и звукозаписей в базу знаний
- Регистрация источника требований для обратной связи







- Дополнение базы знаний по шаблонам RUP
- Автоматическая отправка фотографий и звукозаписей в базу знаний
- Регистрация источника требований для обратной связи
- Подготовка тестовых сценариев и наборов

Сбор
требований



Анализ
требований



Планирование



Разработка

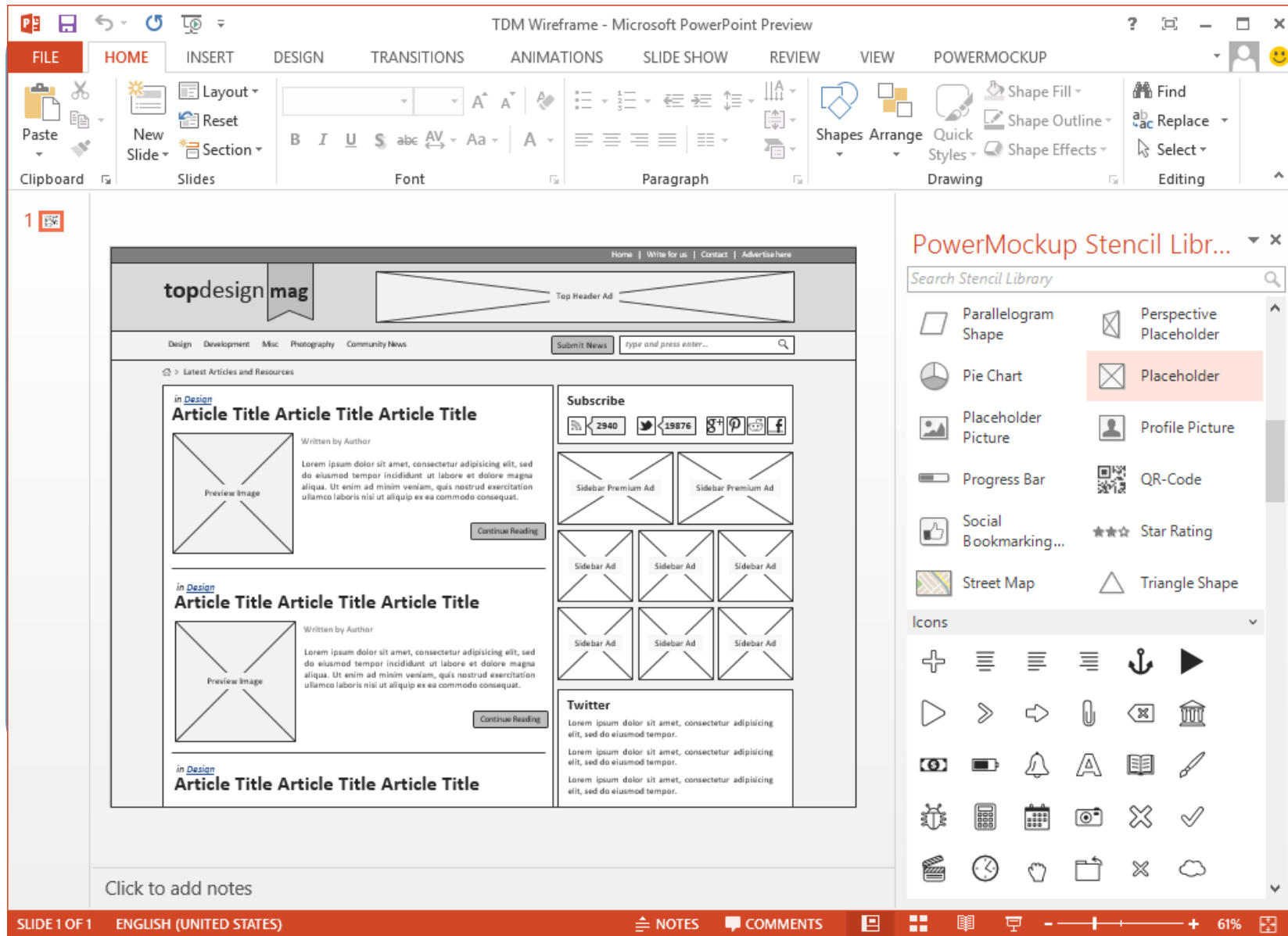


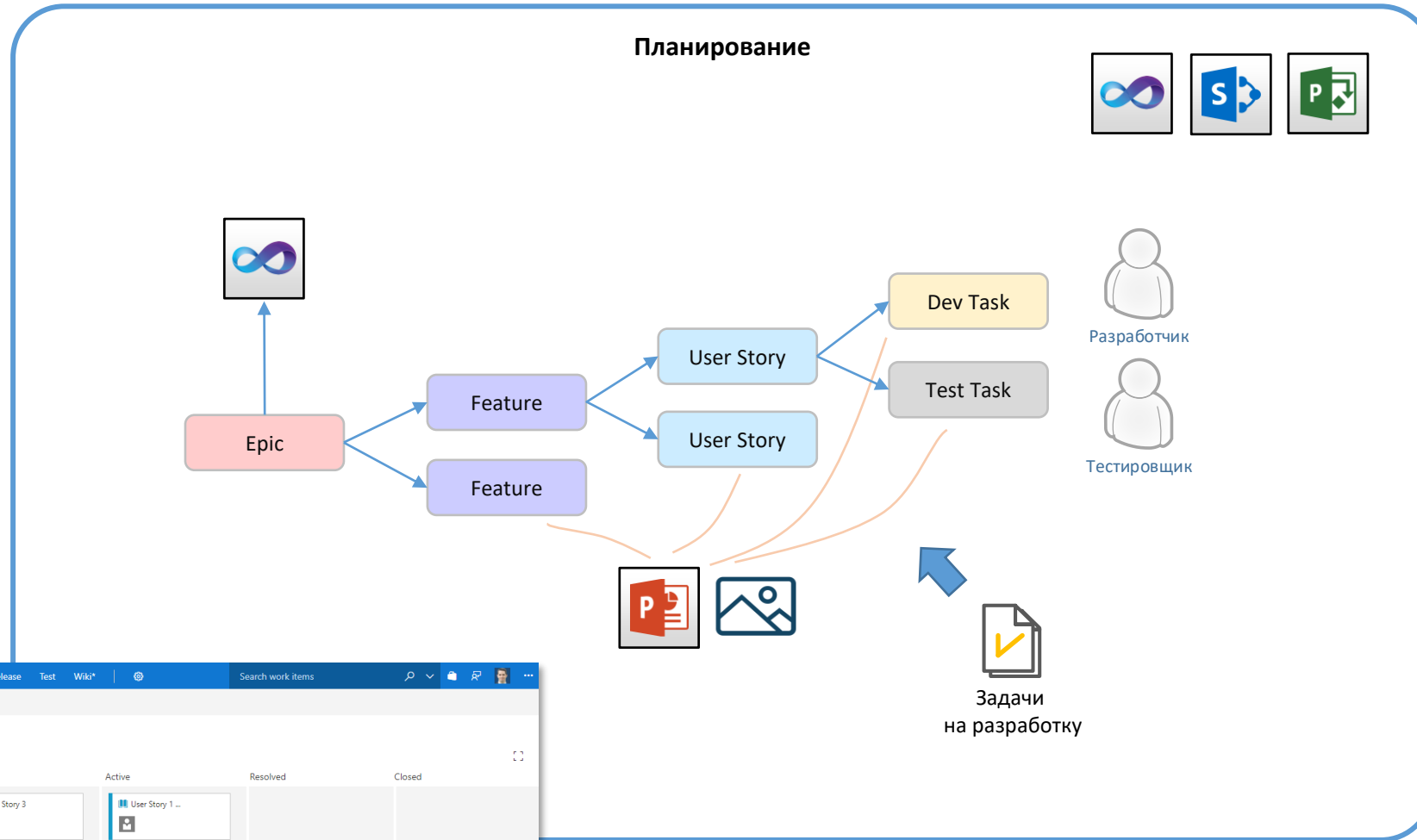
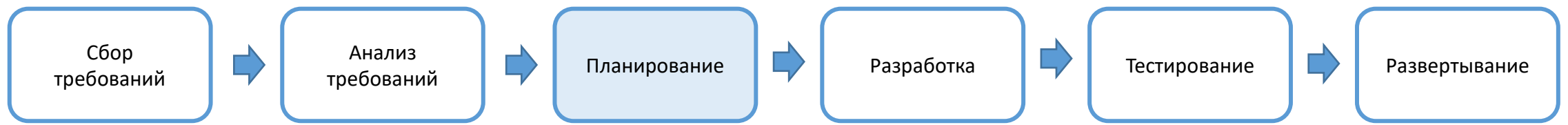
Тестирование



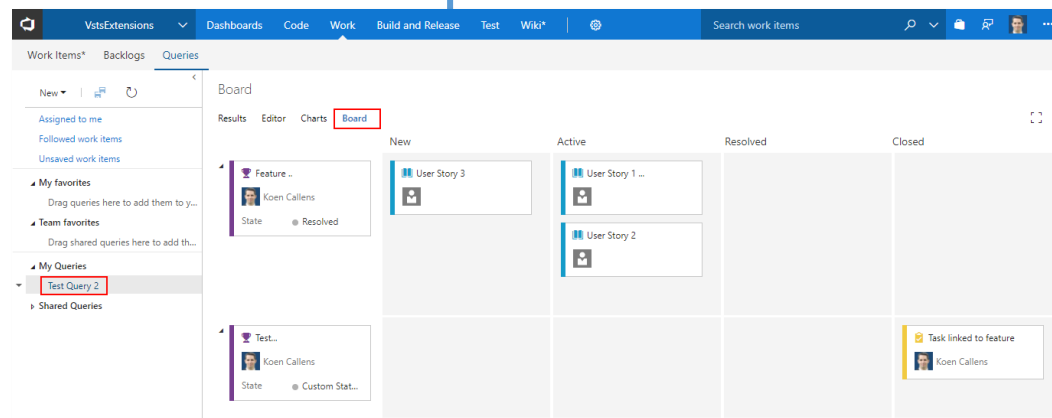
Развертывание

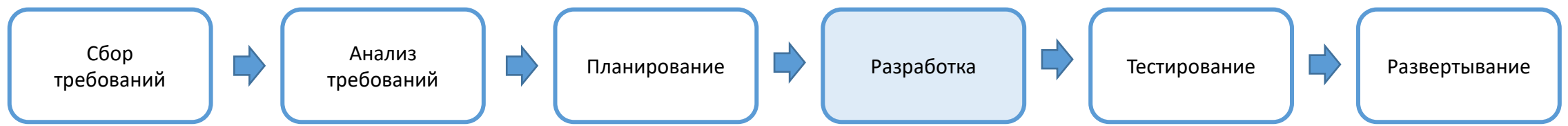
Макетирование UI +
поведения





TFS Scrum Board

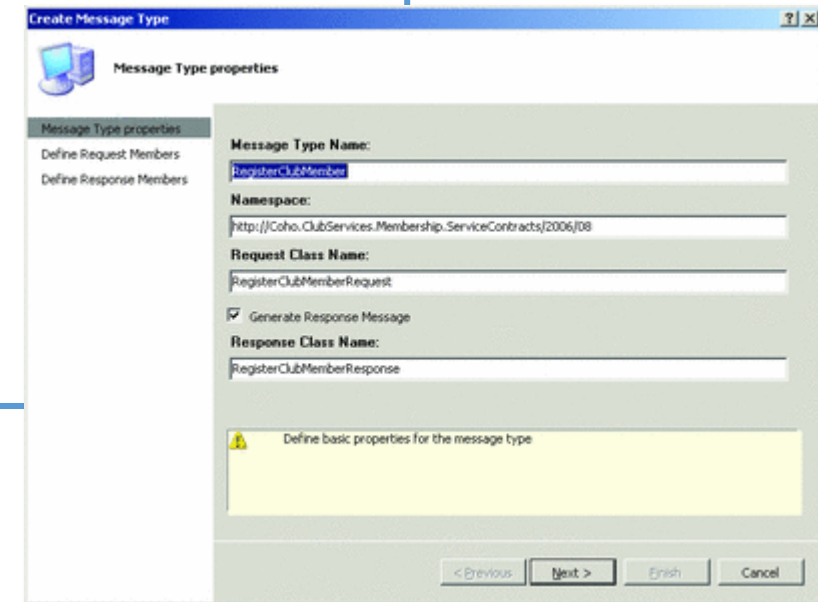
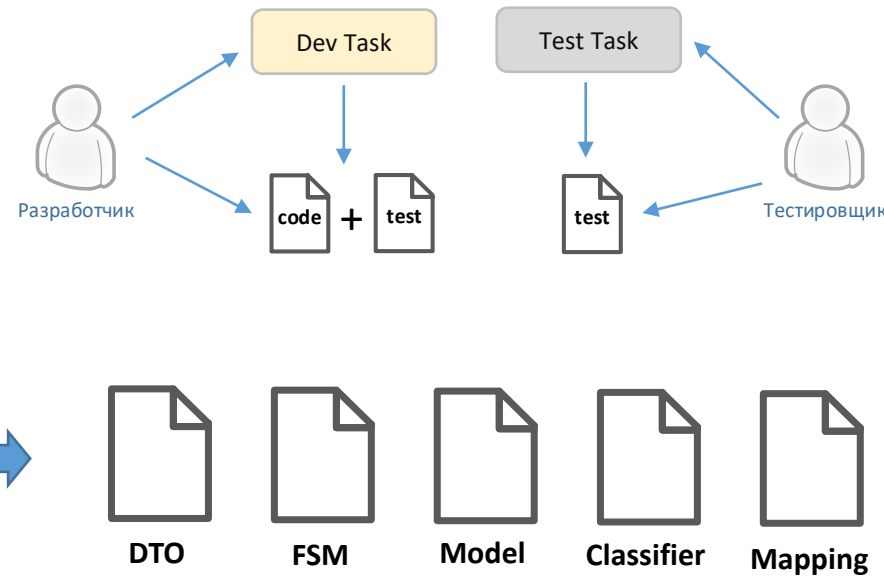
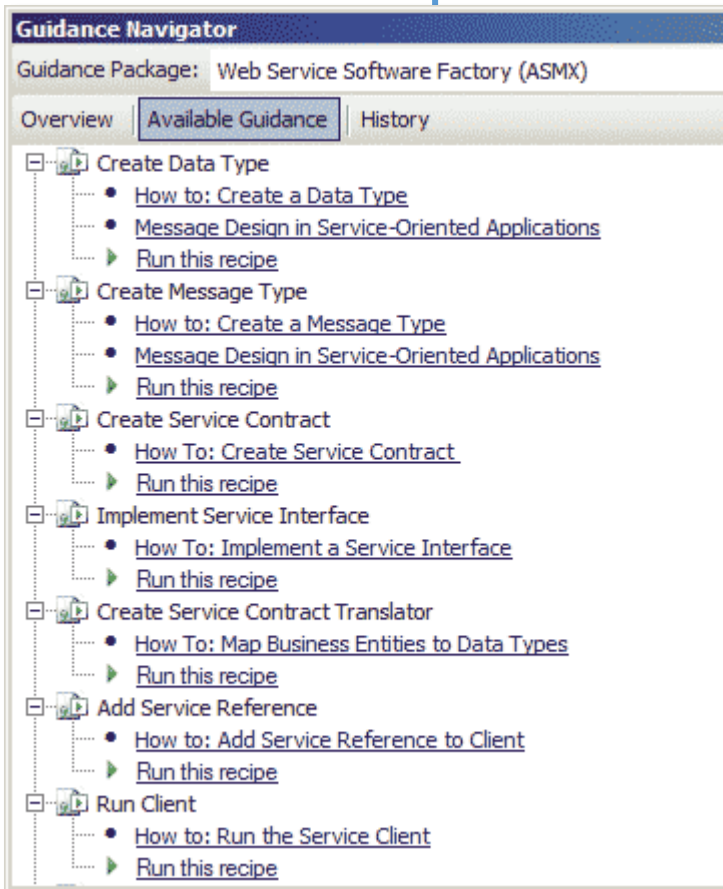


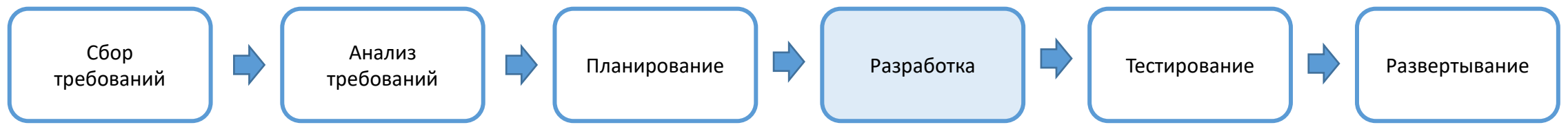


Разработка

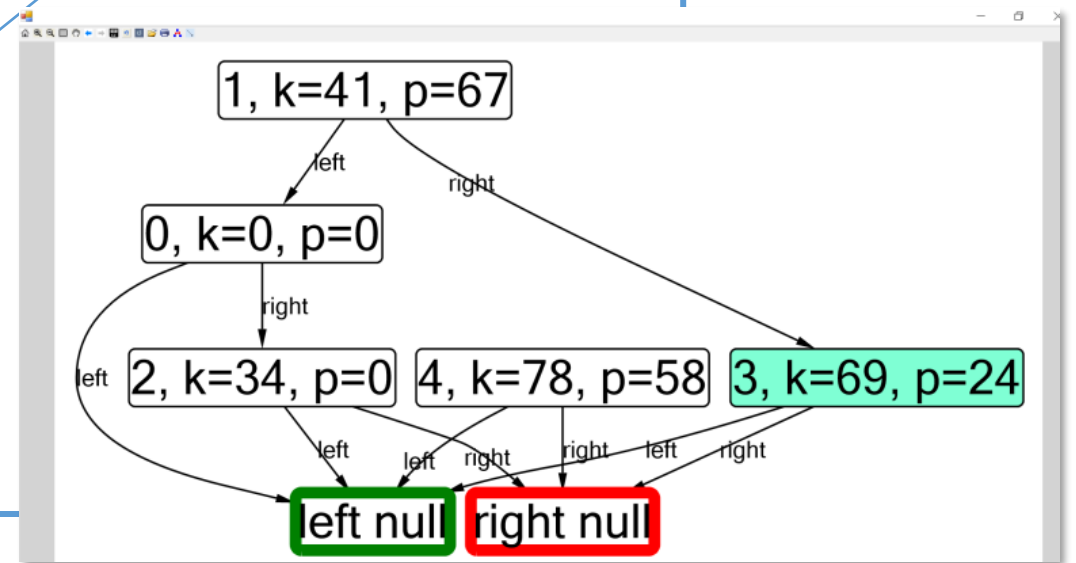
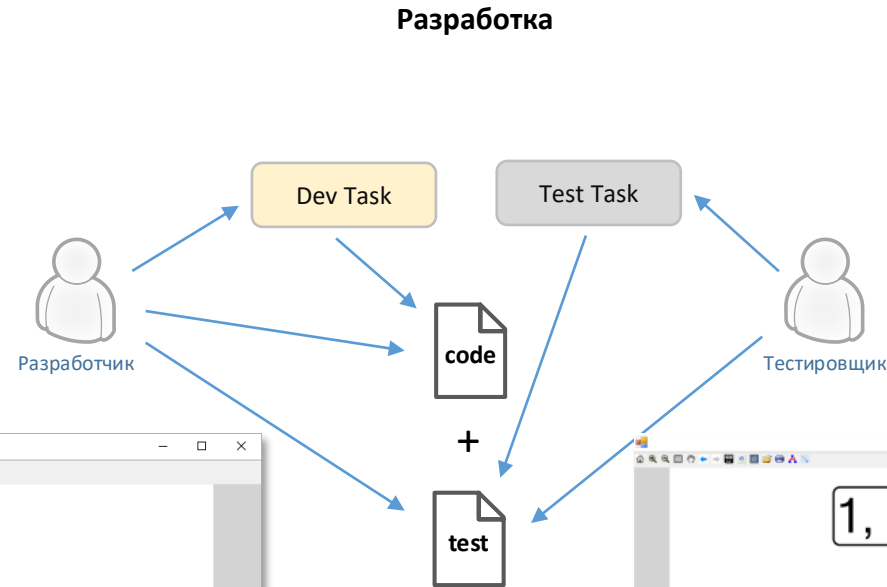
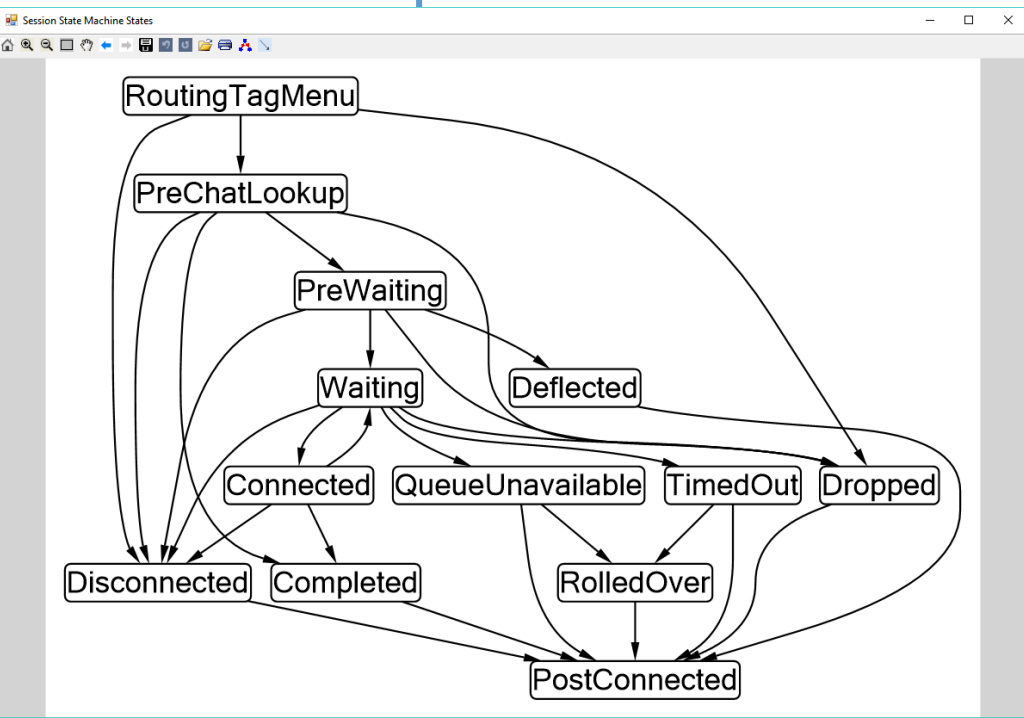


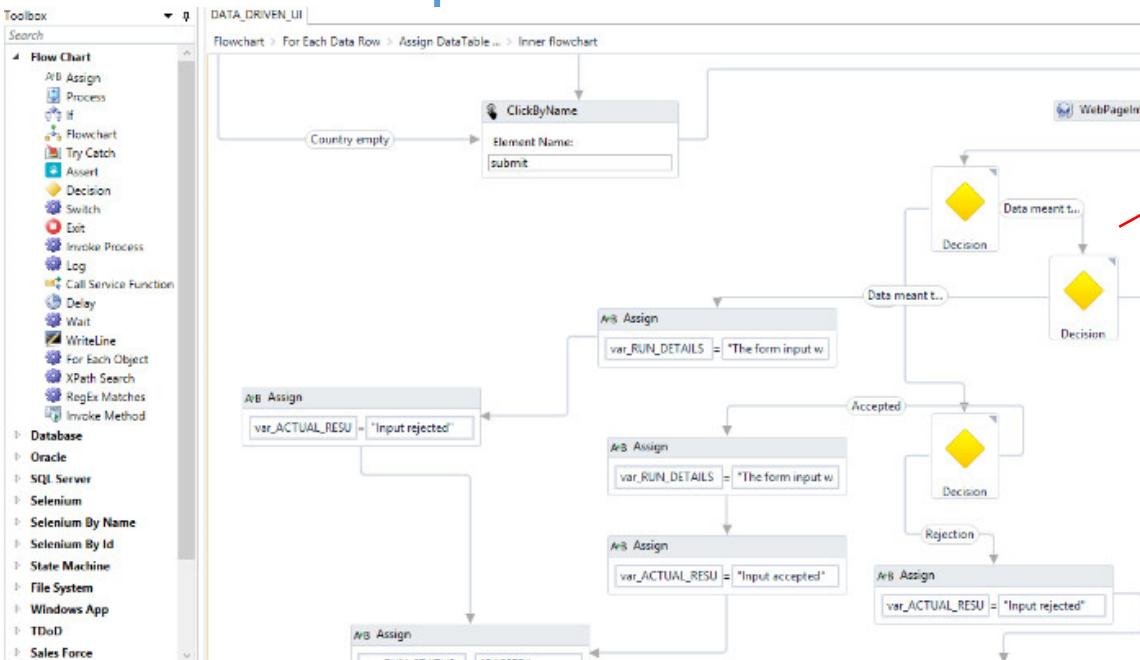
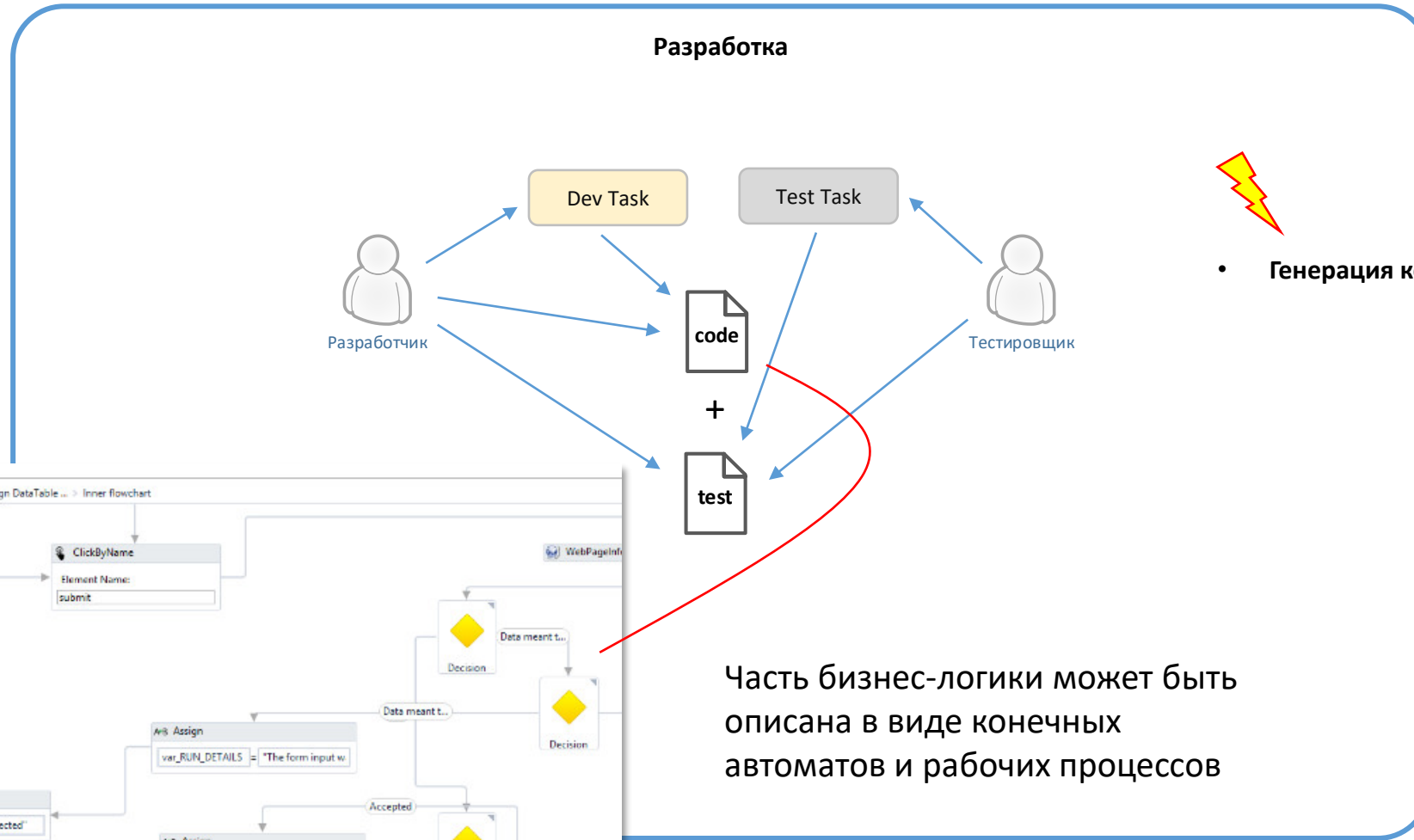
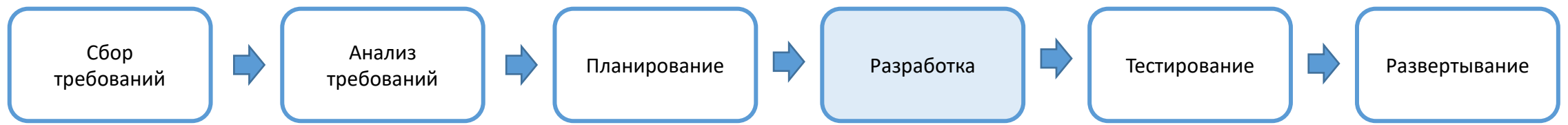
- Генерация доменных классов из UML, Excel, Visio
- Генерация заглушек и клиентов для сервисов по спецификации WSDL
- Генерация DTO, классов-сообщений
- Генерация конфигов для нескольких окружений
- Генерация справочников из БД и Excel
- Генерация конечных автоматов из Excel и UML
- Генерация форм UI

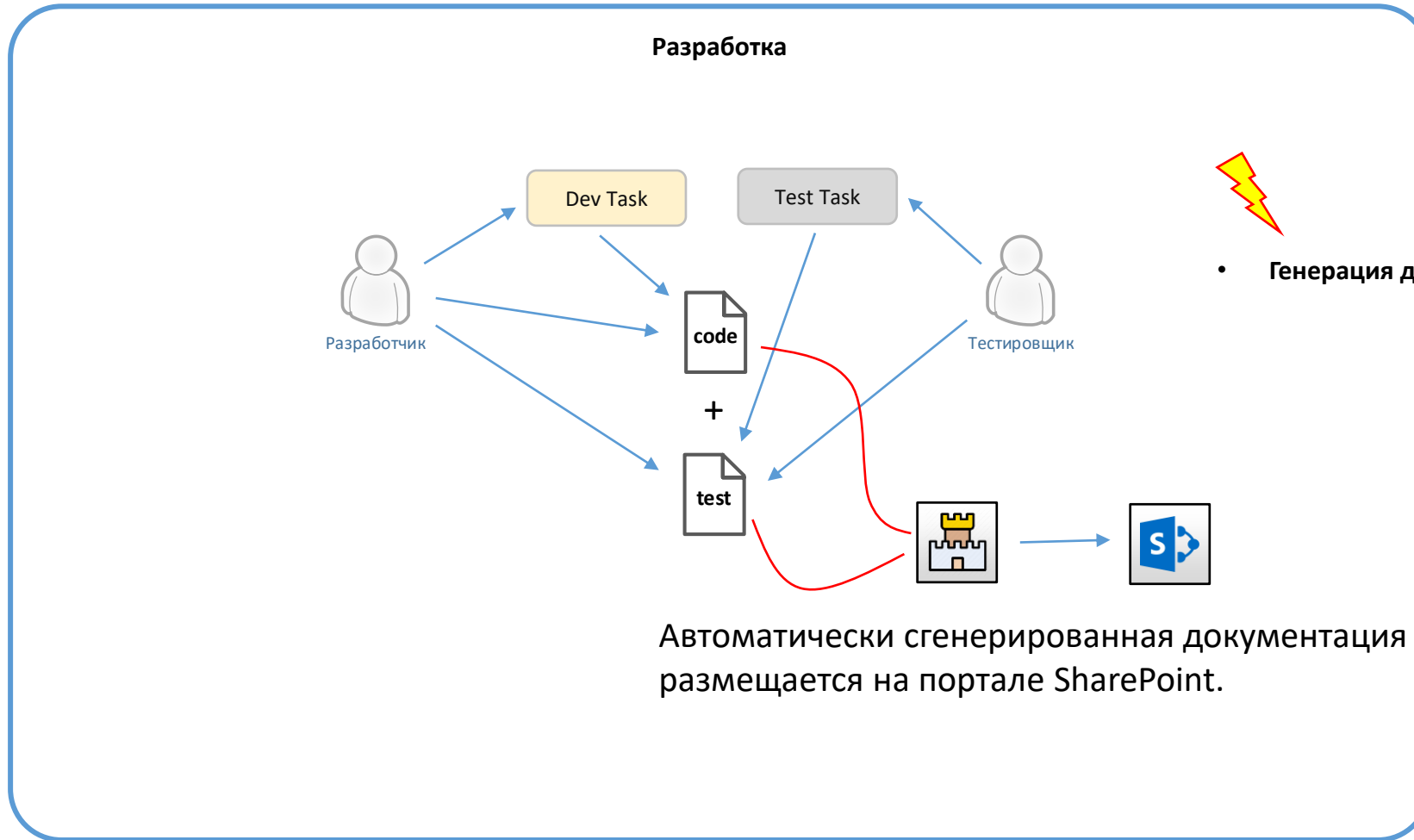
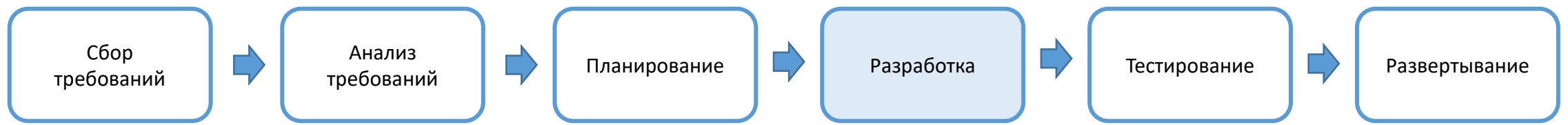


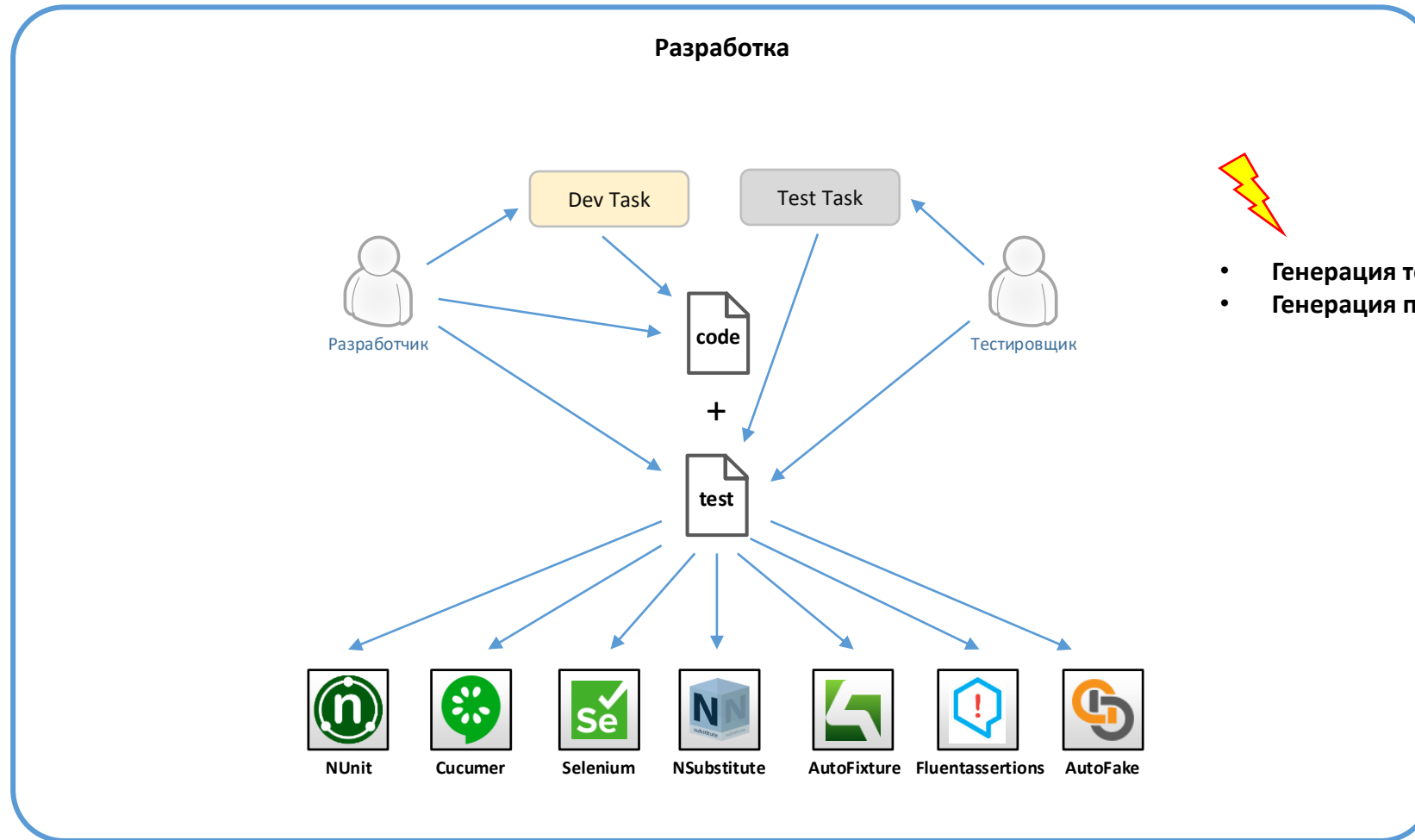
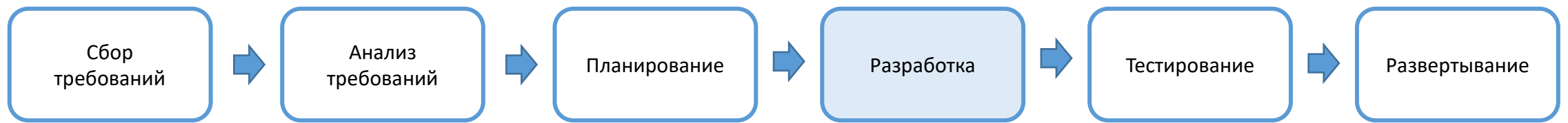


Визуализация важных аспектов выполняющегося кода при помощи расширения отладчика

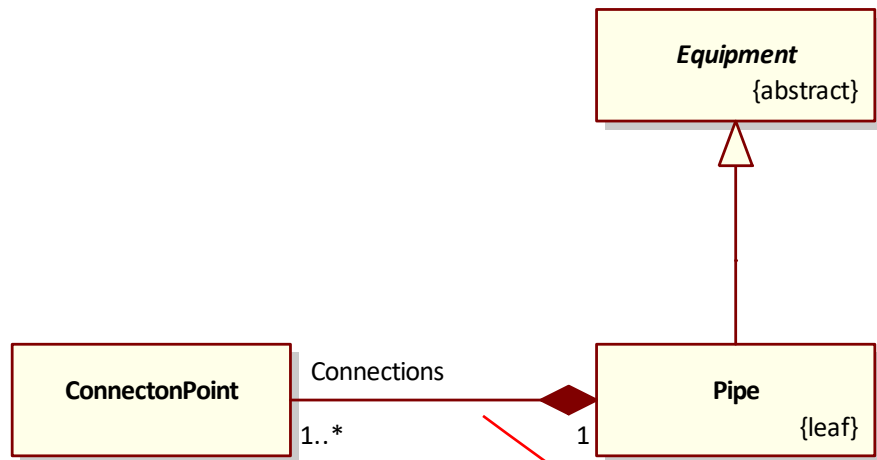








- Фреймворк для модульных тестов
- Фреймворк для приемочных тестов
- Фреймворк для выразительных ассертов
- Инструмент для UI-тестов
- Генератор тестовых заглушек (mock)
- Генератор тестовых данных



```

public sealed partial class Pipe : Equipment
{
    [OneOrMore]
    public IEnumerable<ConnectionPoint> Connections { get; }
}
  
```

DL Template Editor | Code Template Editor

```

template:
1 %if elementType == "Interface"%
2 {\n
3 %list="Operation" @separator="\n\n" @indent="\t"%
4 %PI=""%
5 } //end %className%
6 %endTemplate%
7
8 {\n
9 %PI="\n\n"%
10
11 %list="InnerClass" @separator="\n\n" @indent="\t"%
12 %list="Attribute" @separator="\n\n" @indent="\t"%
13
14 %list="ModuleUsage" @separator="\n\n" @arguments=("class_member_variable_csharp")%
15
16 %list="State" @separator="\n\n" @indent="\t"%
17
18 %if genOptStmCSharpUseNewTemplate != "T" or stmContextHasStateMachine != "T"%
19 %ops=""%
20 %if genOptGenConstructor == "T" and classHasConstructor != "T"%
21 %ops+="%public "+%className%+"() {\n\n\t}"%
22 %endIf%
23
24 %if genOptGenDestructor == "T" and classHasDestructor != "T" and classHasFinalizer != "T"%
25 %ops+="%\n\n\t"+%className%+"() {\n\n\t}"%
26 %endIf%
27 %endIf%
28
29 %if genOptCSGenDispose == "T" and classHasDispose != "T"%
30 %ops+="%\n\n\tpublic "
31 %ops+="%classHasParent=="T" ? "override " : "virtual "%
32 %ops+="%void Dispose() {\n\n\t}"%
33 %endIf%
34
35 %if genOptCSGenFinalizer == "T" and genOptGenDestructor != "T" and classHasFinalizer != "T" and classHasDestructor != "T"%
36 %ops+="%\n\n\tprotected "
37 %ops+="%classHasParent=="T" ? "override " : "virtual "%
38 %ops+="%void Finalize() {\n\n\t}"%
39 %ops+="%classHasParent=="T" ? "\t\tbase.Finalize(); : ""% + "\n\n\t}"%
40 %endIf%
41
42 %if genOptGenCopyConstructor == "T" and classHasCopyConstructor != "T"%
43 %ops+="%\n\n\tpublic "
44 %ops+="%className%+"("+%className%+" the"+%className%+" ) {\n\n\t}"%
  
```

Get Default Template | Save | Delete | Help

All Perspectives | CAP | NUM | SCRL | CLOUD

Трансляция UML-модели в классы предметной области с помощью шаблонов кодогенерации.

Дополнительные ограничения модели представлены в виде специальных аннотаций.

Test Explorer

Streaming Video: Improving quality with unit tests and fak...

Run All | Run... | Playlist : All Tests

CustomerListTests (7)

✗ DeleteCustomer_ShouldRemoveCustomerFromList	19 ms
✓ AddCustomer_InvalidDataShouldSetErrorMessage...	< 1 ms
✓ AddCustomer_ValidDataShouldAddCustomerTo...	< 1 ms
✓ AddCustomer_ValidDataShouldReturnCustomer...	< 1 ms
✓ AddCustomer_ValidDataShouldSendEmailToCus...	< 1 ms
✓ EditCustomer_InvalidDataShouldSetErrorMessage	< 1 ms
✓ EditCustomer_ValidDataShouldModifyCustomer...	< 1 ms

DeleteCustomer_ShouldRemoveCustomerFromList

Source: CustomerListTests.cs line 40

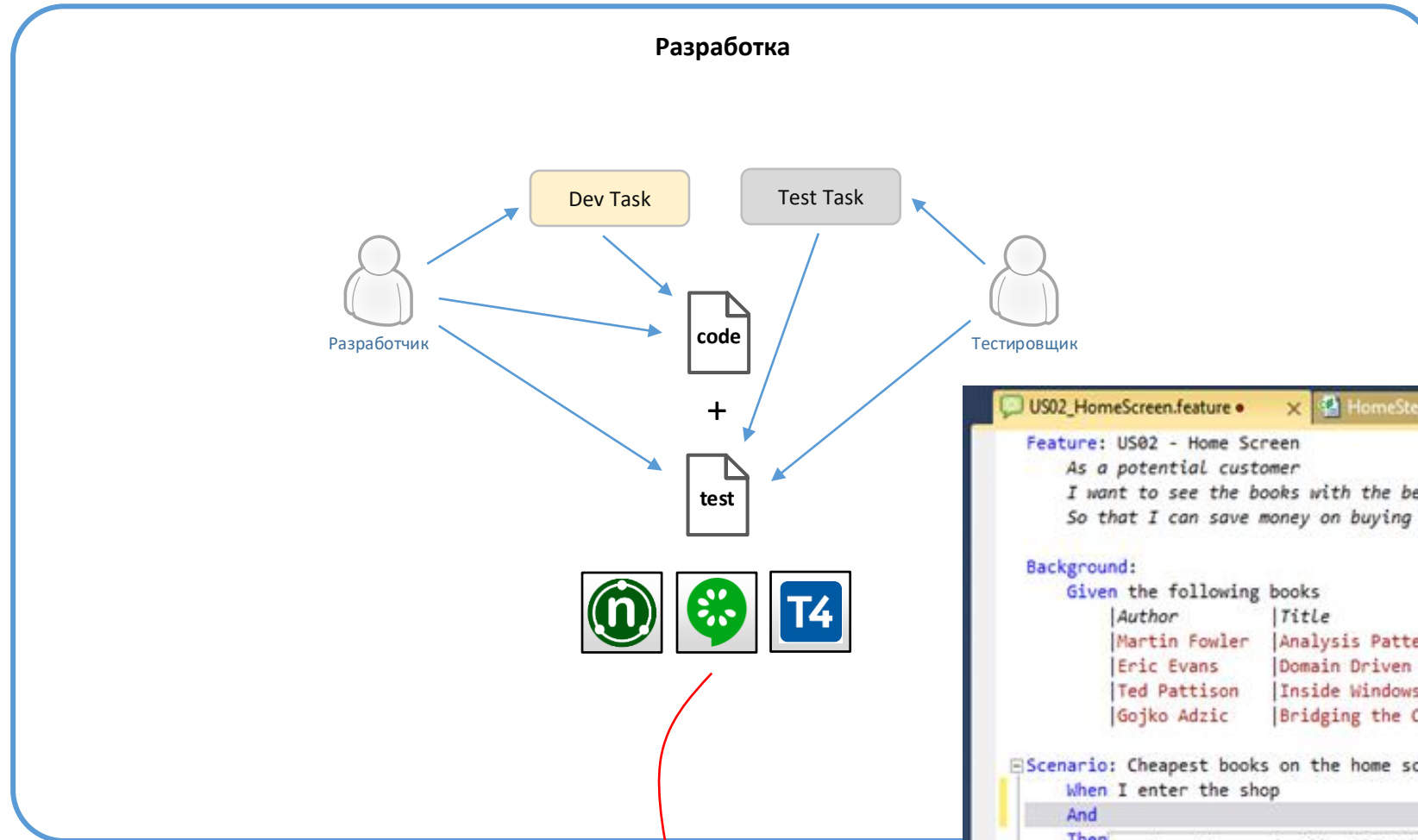
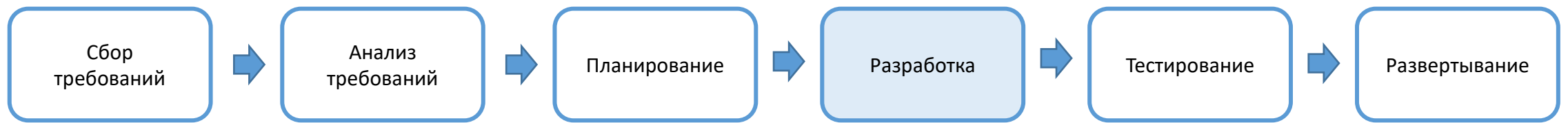
✗ Test Failed - DeleteCustomer_ShouldRemoveCustomer

Message: Assert.IsTrue failed.

Elapsed time: 19ms

Stack Trace:

CustomerListTests.DeleteCustomer_ShouldRem...



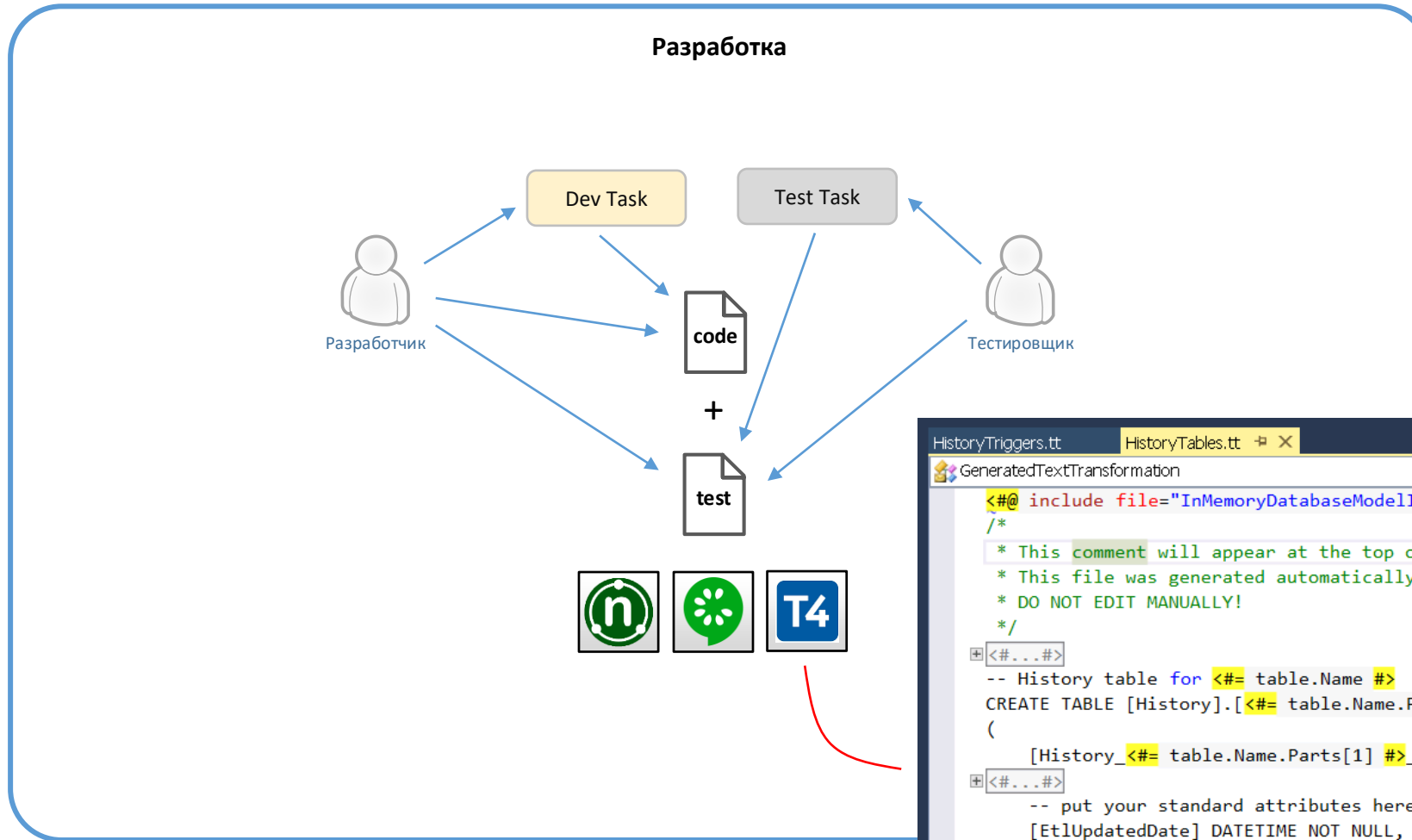
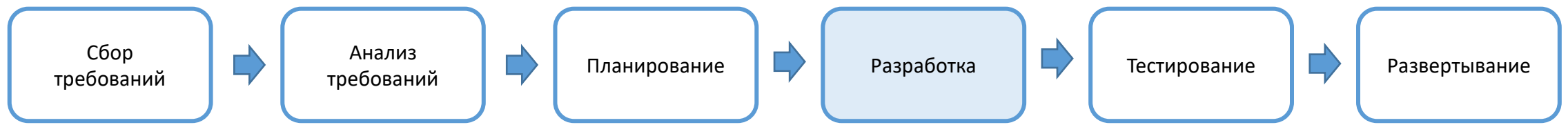
Приемочные тесты в
стиле BDD на Cucumber

US02_HomeScreen.feature • HomeSteps.cs

```
Feature: US02 - Home Screen
  As a potential customer
  I want to see the books with the best price
  So that I can save money on buying discounted books.

Background:
  Given the following books
    | Author | Title | Price |
    | Martin Fowler | Analysis Patterns | 50.20 |
    | Eric Evans | Domain Driven Design | 46.34 |
    | Ted Pattison | Inside Windows SharePoint Services | 31.49 |
    | Gojko Adzic | Bridging the Communication Gap | 24.75 |

Scenario: Cheapest books on the home screen
  When I enter the shop
  And
  Then
  And
  And
  @alterna
  Scenario
  When
  And
```

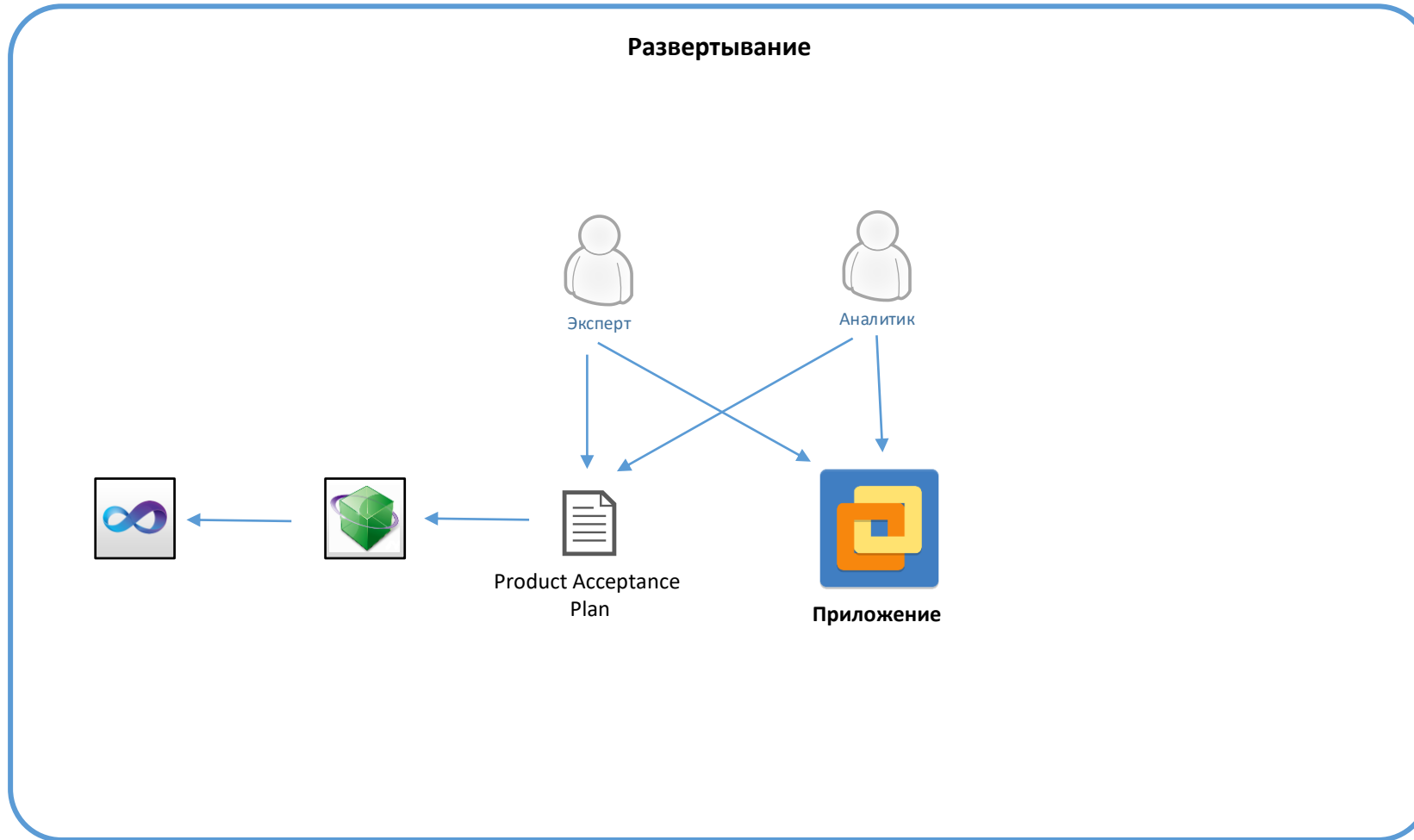
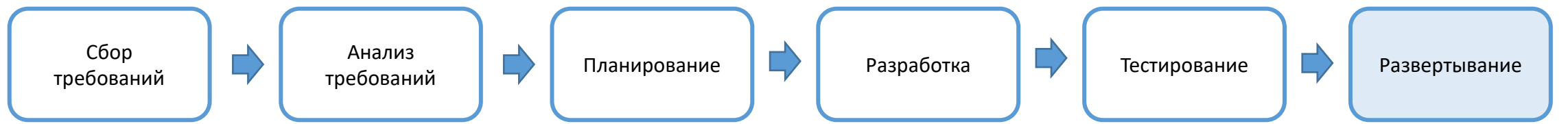


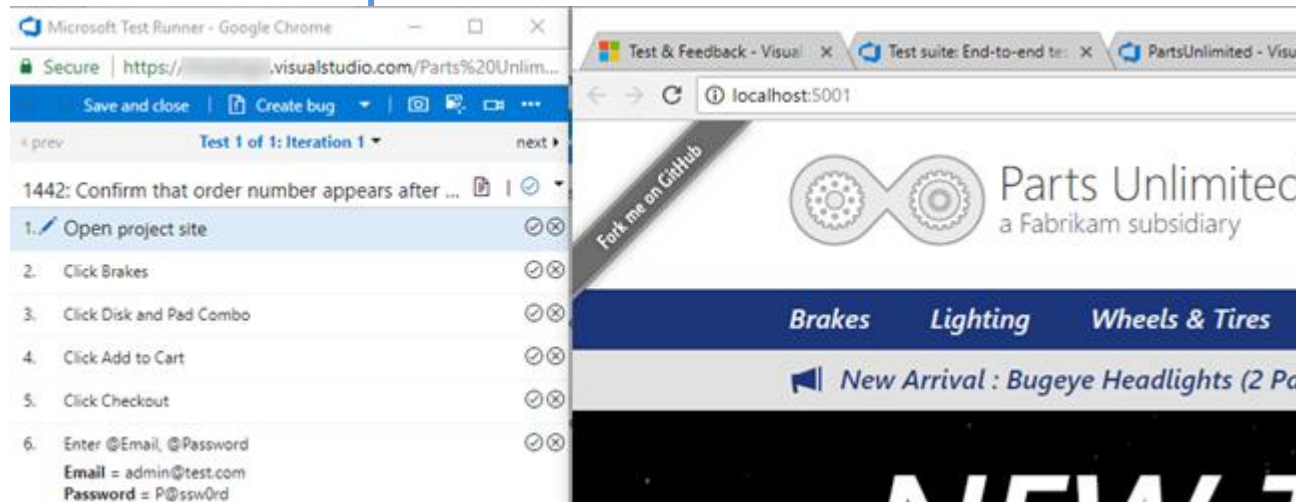
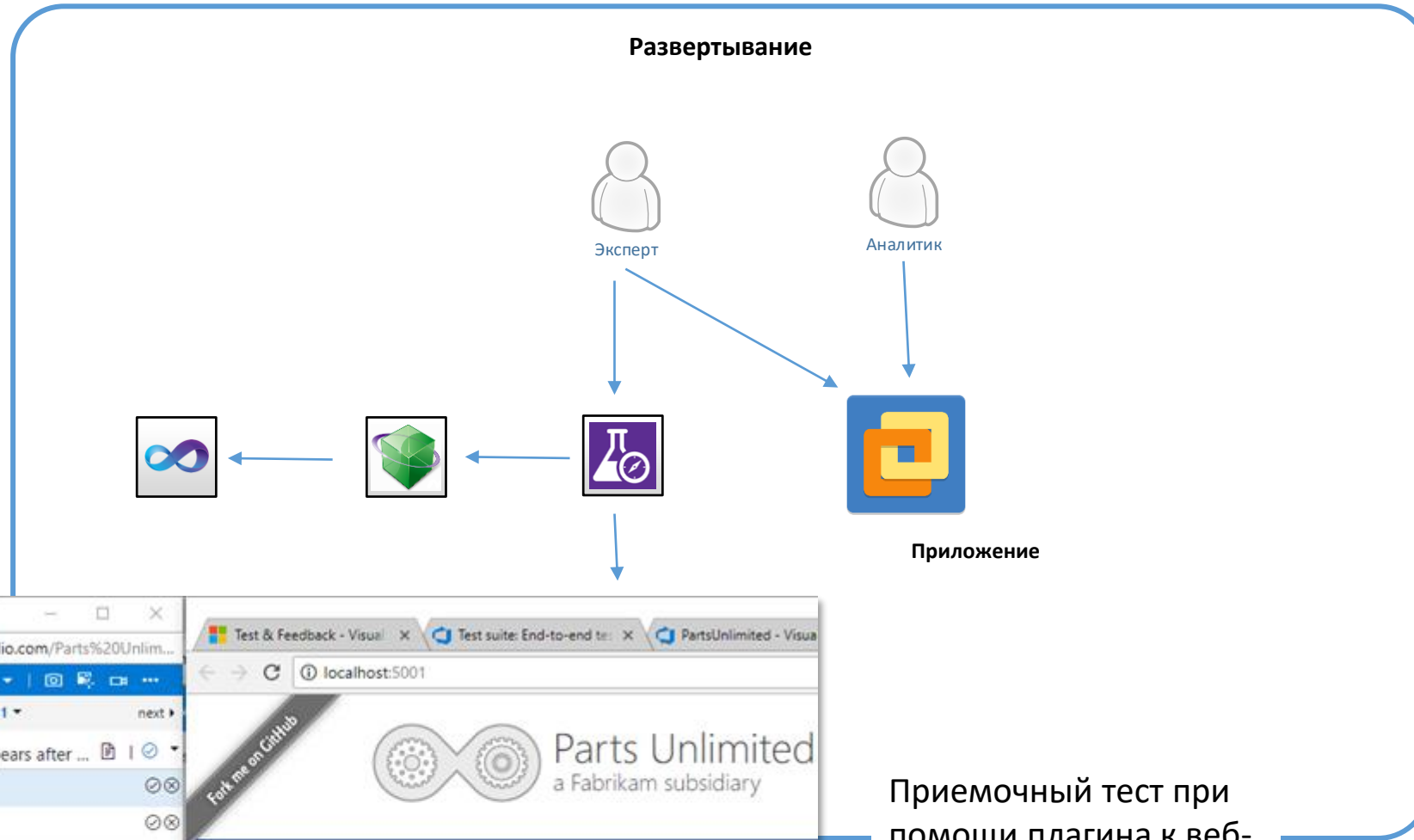
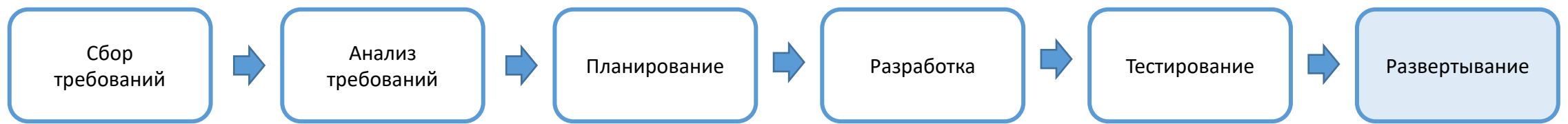
```
HistoryTriggers.tt  HistoryTables.tt  X
GeneratedTextTransformation

<#@ include file="InMemoryDatabaseModelInclude.tt" #>
/*
 * This comment will appear at the top of your auto-generated file.
 * This file was generated automatically by a T4 Template
 * DO NOT EDIT MANUALLY!
 */

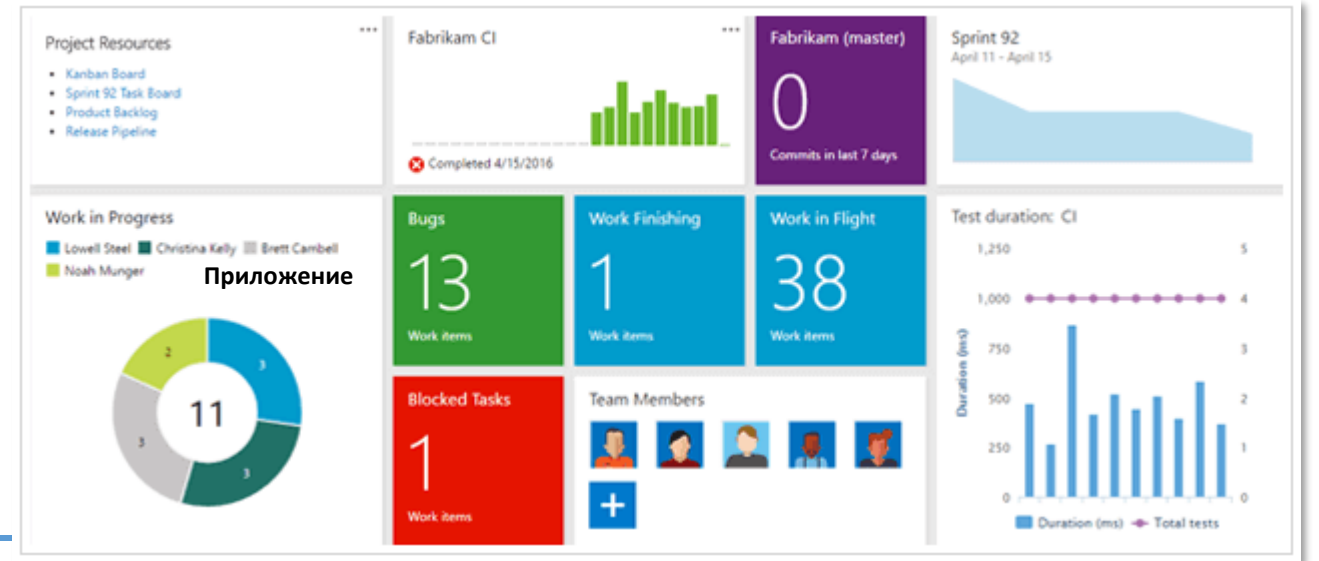
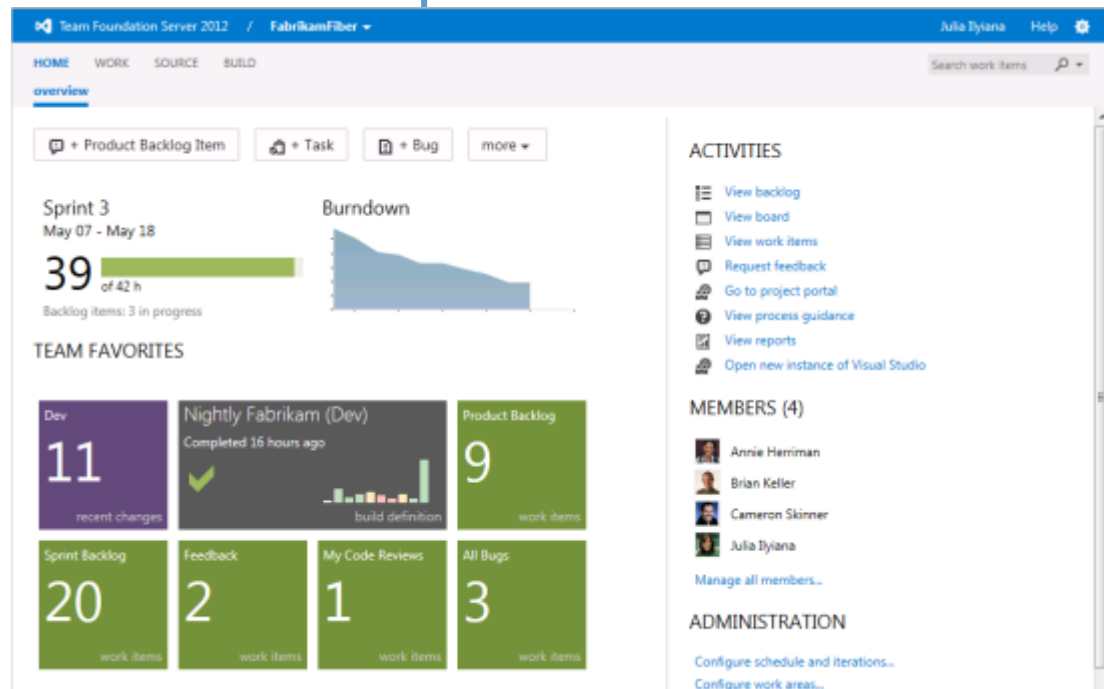
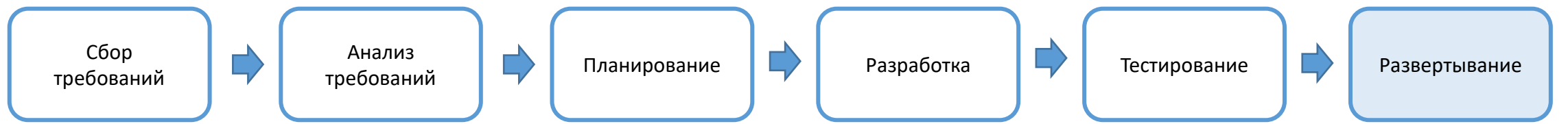
<#...#>
-- History table for <#= table.Name #>
CREATE TABLE [History].[<#= table.Name.Parts[1] #>]
(
    [History_<#= table.Name.Parts[1] #>_ID] INT NOT NULL IDENTITY(1,1),
    -- put your standard attributes here
    [Et1UpdatedDate] DATETIME NOT NULL,
) ON [History];
GO

<#...#>
-- End of auto generated code!
```



Приемочный тест при помощи плагина к веб-браузеру.



Выводы

- Была ли устранена изначальная проблема?
- Что в итоге получилось и не получилось?
- Стоимость внедрения

Вопросы?



Software Factory

- Генерация доменной модели из UML и других источников (XML, JSON, Excel, GraphML)
- Генерация кода по шаблонам (T4, Mustache)
- Расширения отладчика (визуализаторы моделей, графов, структур данных)
- Генераторы – визарды
- Шаблоны проектов
- Шаблоны кода и артефактов
- База знаний, интегрированная с IDE
- Автогенерация документации
- Готовые тестовые наборы для проверки спецификаций и стандартов
- BDD, Cucumber как приемочный тест
- Наборы нагрузочных тестов, производительности, стресс (для проверки НФТ)
- Готовые генераторы тестовых данных
- Плагины для IDE (рефакторинг, валидация, генерация)
- Шаблоны документов, примеры, инструкции
- Генератор моделей Visio
- Библиотека виджетов + инструмент тестирования
- Анализаторы кода (проверка модели, арх.ограничения)