Case-системы

Автор:
студент группы ВМИ-356
М.Е. Кутырева

Проверил:
Кандидат физ.-мат. наук, доцент,
Г.И. Радченко

Содержание

- Что такое case-системы?
- Виды, типы, категории case-систем
- Системы управления требованиями
- Continuous Integration
- IBM Software Architect
- Системы коллективной разработки ПО

Case-системы (Computer-Aided Software Engineering) - средства разработки программных и организационно-управляющих систем.

Зачем их использовать?

- ошибок становится меньше;
- качество ПО выше;
- обслуживание проще.

До…

После =)

Case-системы:

- по типам – отражает внутреннюю ориентацию на те или иные процессы ЖЦ;
- по категориям – степень интегрирования по выполняемым функциям.

Классификация по типам:

- репозиторий, являющийся основой CASE-средства;
- графические средства анализа и проектирования;
- средства разработки приложений;
- средства конфигурационного управления;
- средства документирования;
- средства тестирования;
- средства управления проектом;
- средства реинжиниринга.

Классификация по категориям:

- *tools* - вспомогательные программы;
- *toolkit* - пакеты разработки;
- *workbench* - инструментальные средства.

Классификация по области действия в ЖЦ ПО:

- Upper CASE;
- Middle CASE;
- Lower CASE;

Дополнительная (объединенная) классификация (по использованию):

- анализ и тестирование;
- для проектирования баз данных и файлов;
- для процесса реализации;
- для процесса внедрения;
- для сопровождения и реинженерии;
- для управления проектом.

Системы управления требованиями

- **Управление требованиями –** процесс, включающий идентификацию, выявление, документацию, анализ, отслеживание, приоретизацию требований, достижение соглашений по требованиям, управление этими изменениями
- **Требование:**
  - условие или возможность, необходимые для решения проблем или достижения целей;
  - условие или возможность, которыми должна обладать система или системные компоненты, чтобы выполнить контракт или удовлетворить каким-либо стандартным документам;
  - документирование вышеописанных пунктов.


Выявление требований
Анализ требований
Управление требованиями
- управление изменениями
- назначение атрибутов
- трассировка требований
Утверждение требований
Спецификация требований

Характеристики требований

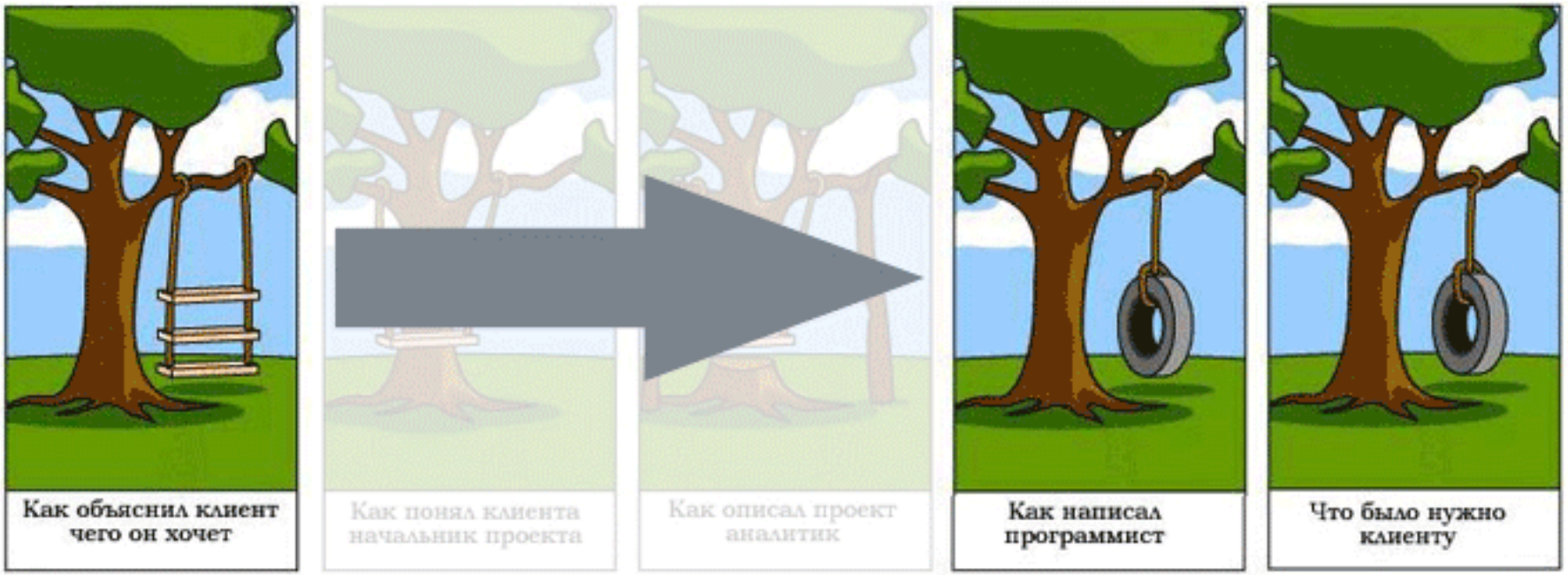
- единичность;
- завершенность;
- последовательность;
- атомарность;
- отслеживаемость;
- актуальность;
- выполнимость;
- недвусмысленность;
- обязательность;
- проверяемость.

Группы требований

- *Функциональные* – реализация самой бизнес-функции;
- *Управленческие* – администрирование, безопасность, размещение;
- *Эргономические* – удобство пользователя;
- *Архитектурные* – к архитектуре системы;
- *Взаимодействия* – существующие+новые
- *Сервисного уровня* – поведение сервиса, качество входных/выходных данных и т.д.

Топ-системы наших дней

- IBM Rational RequisitePro;
- IBM Rational /Telelogic DOORS;
- Borland Caliber RM.


Как объяснил клиент чего он хочет
Как понял клиента начальник проекта
Как описал проект аналитик
Как написал программист
Что было нужно клиенту

Continuous Integration

- **Continuous Integration** – практика разработки ПО, заключающаяся в выполнении часых автоматизированных сборок проекта для скорейшего выявления и решения проблем интеграции.
- Один из приемов экстремального программирования

Если программисты разрабатывают независимо друг от друга => интеграция – конечная стадия => работа может **внезапно** затянуться

Continuous Integration - снижение трудоемкости интеграции, раннее обнаружение ошибок и недочетов

5 шагов внедрения **Continuous Integration**

- Реорганизация кода
  - Быстрая сборка
  - Возможность прогнать авто-тесты
- Настройка автоматической системы сборки
- Хранение всех рабочих версий продукта
- Тесты (наличие рабочего продукта)
- Соблюдение процесса разработки


Docs
CI
Build Automation
Project Management
Client
auto make
Bug Tracker
TESTS
C++

IBM Software Architect

Почему это не боль, а радость

- разработка требований (SysML, UML);
- прослеживаемость требований;
- совместная работа в группах;
- визуальная разработка;
- поддержка ЖЦ ПО, интеграции с другими IBM Rational и прочие плюшки.

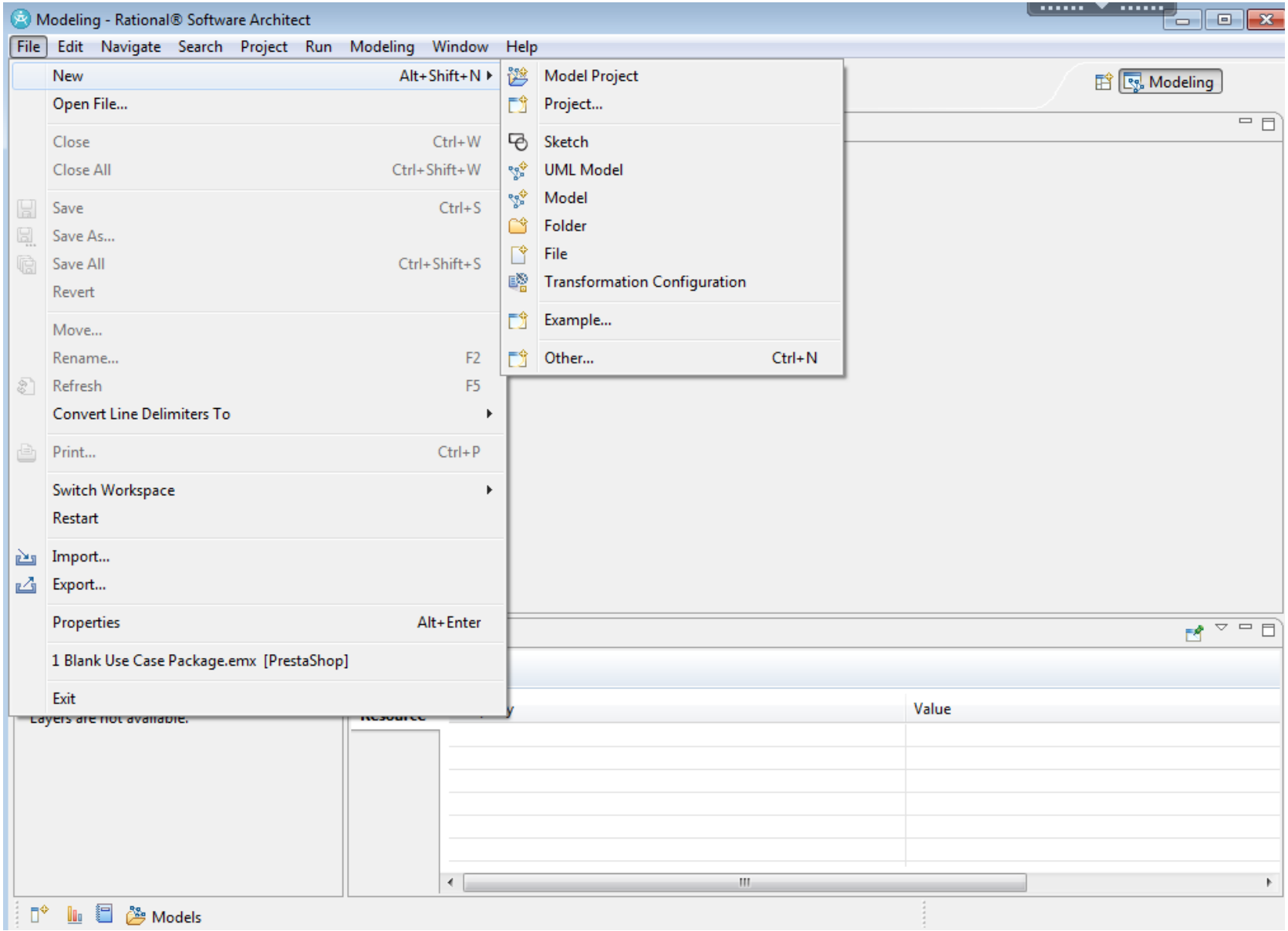
Задача: разработать приложение

Как будем это делать?

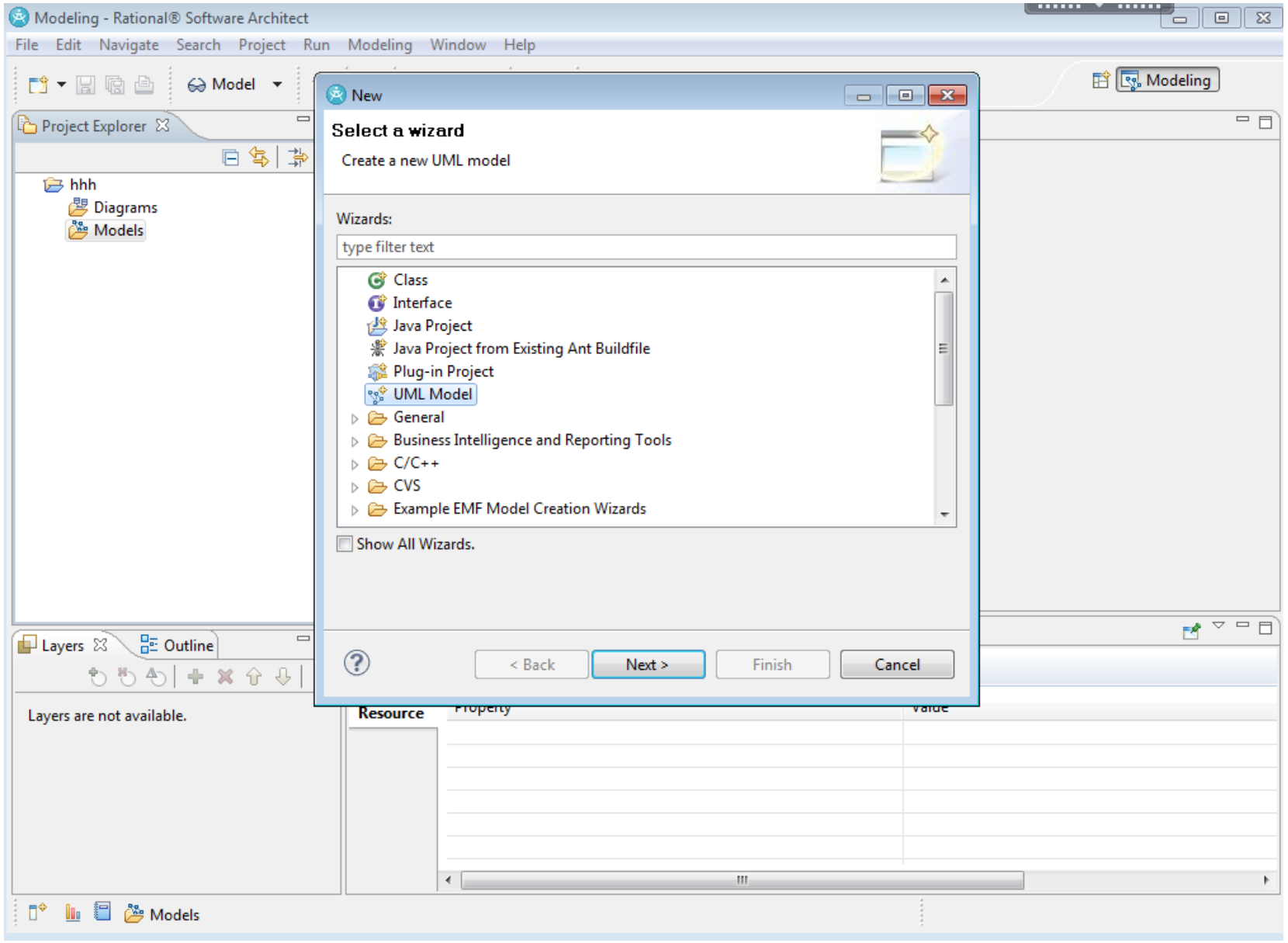
- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- опубликование проекта;
- непосредственное преобразование UML в Java.

- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- построение диаграммы последовательностей;
- опубликование проекта;
- непосредственное преобразование UML в Java.

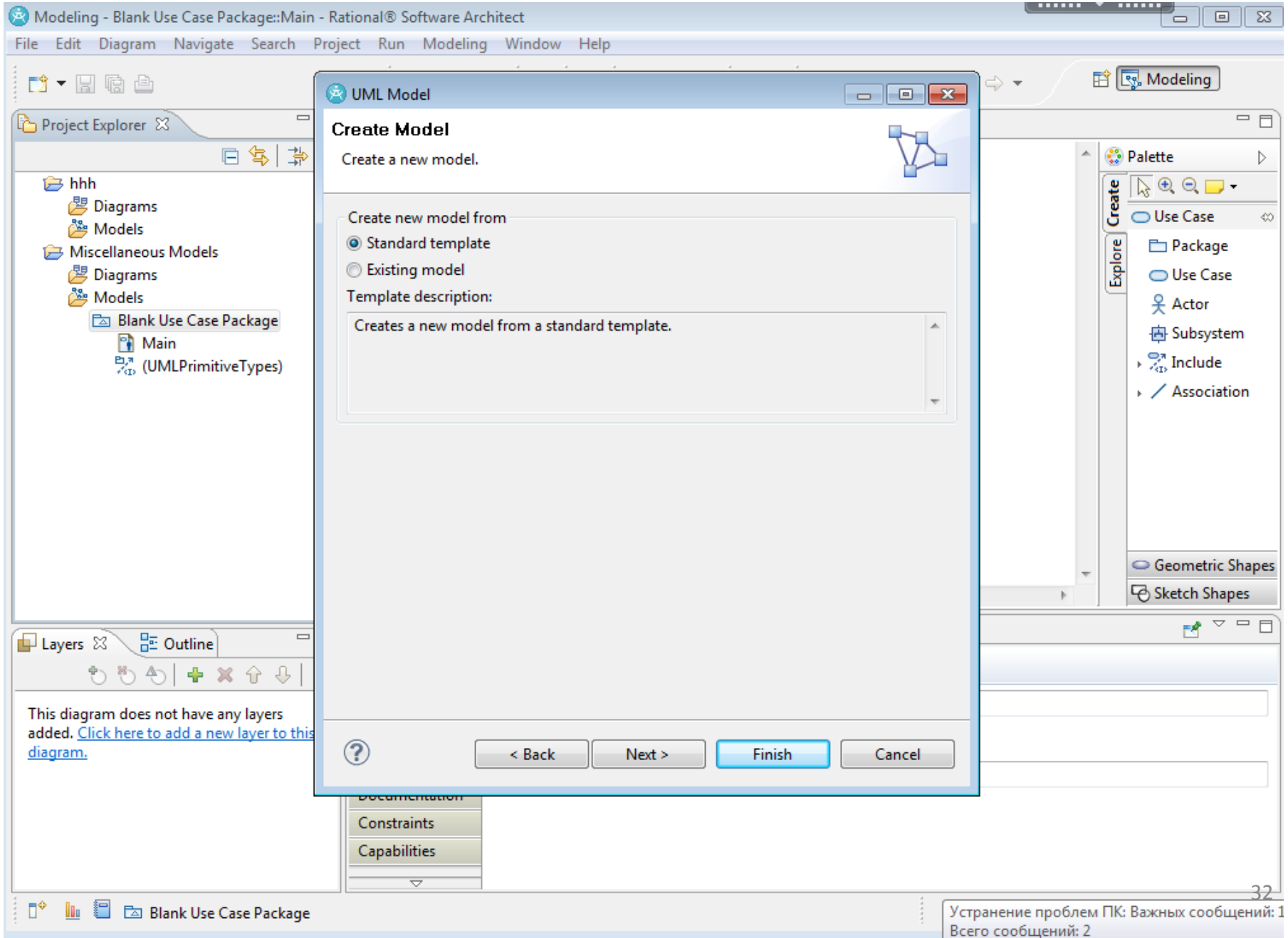
Меню> File > New > Project > Other.



Выбираем UML Project и щелкаем на Next



Выбираем какая должна быть модель и щелкаем на Finish

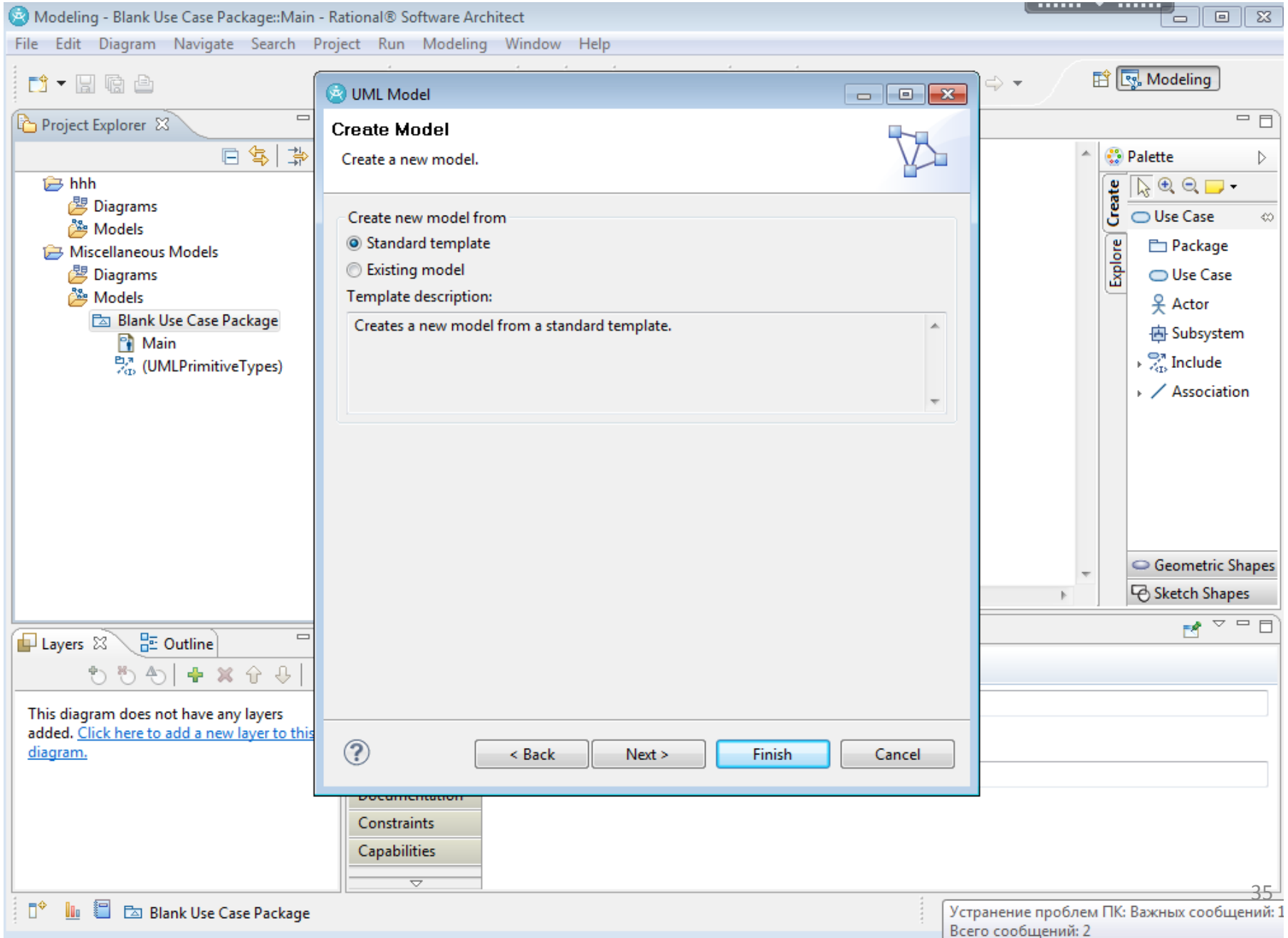


- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- построение диаграммы последовательностей;
- опубликование проекта;
- непосредственное преобразование UML в Java.

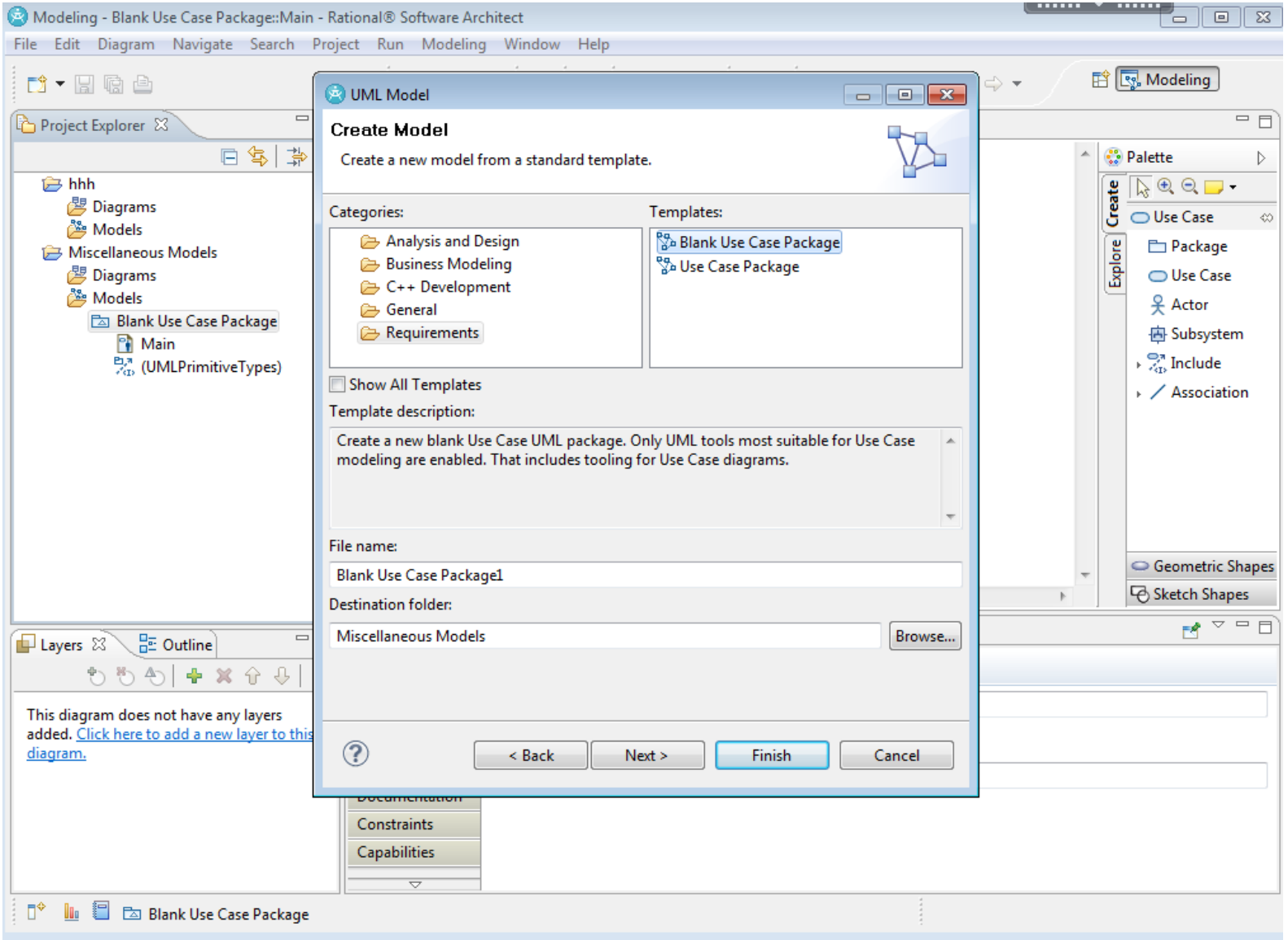
Как сделать магию?

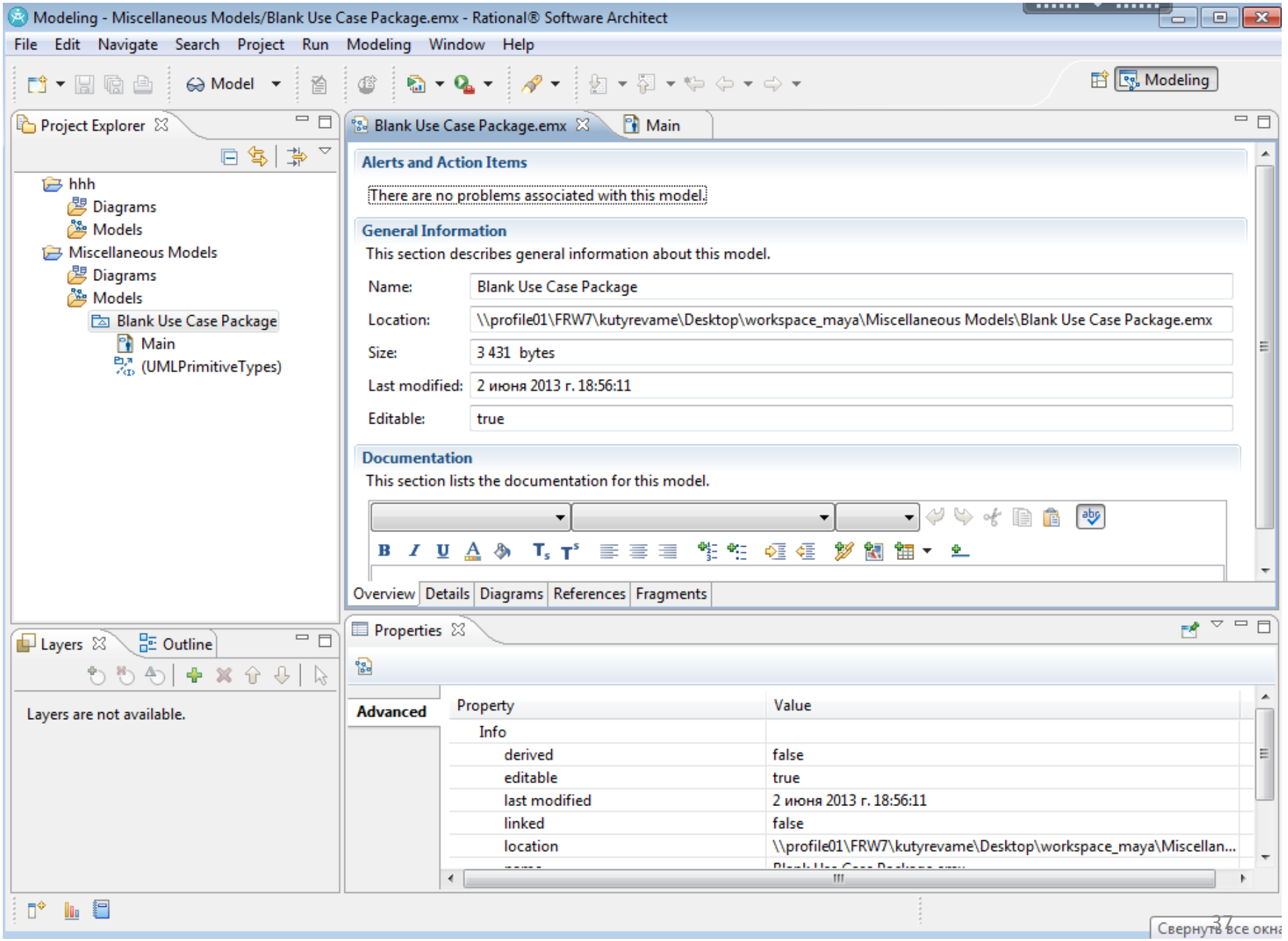
1. выбираем вкладку Model Explorer кликаем правой кнопкой на ранее созданный проект;
2. если его нет выполняем алгоритм «создание UML проекта»;

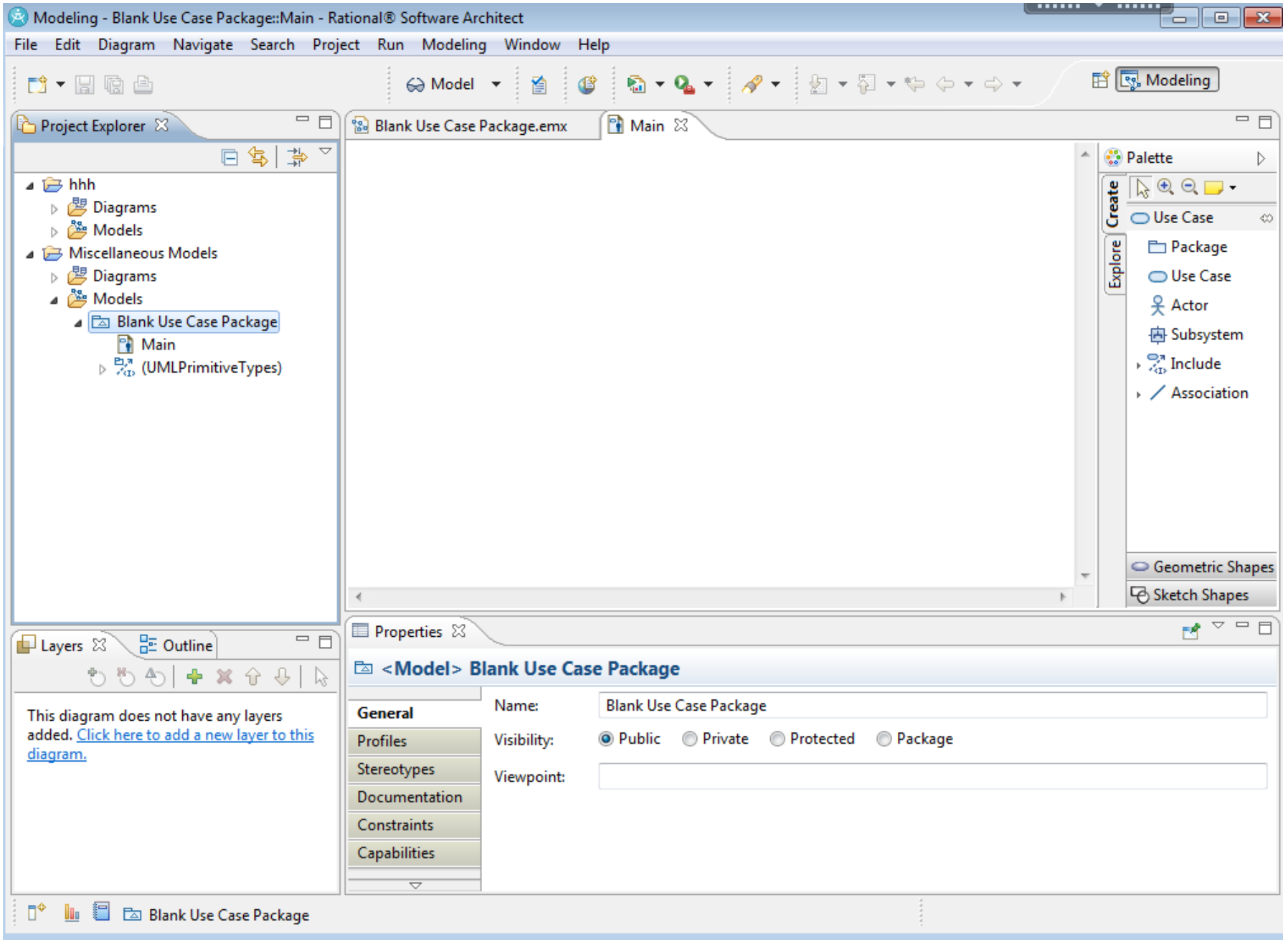
Выбираем какая должна быть модель и щелкаем на Next



В контекстном меню выбираем Requirements> Blank Use Case Package, жмем Finish

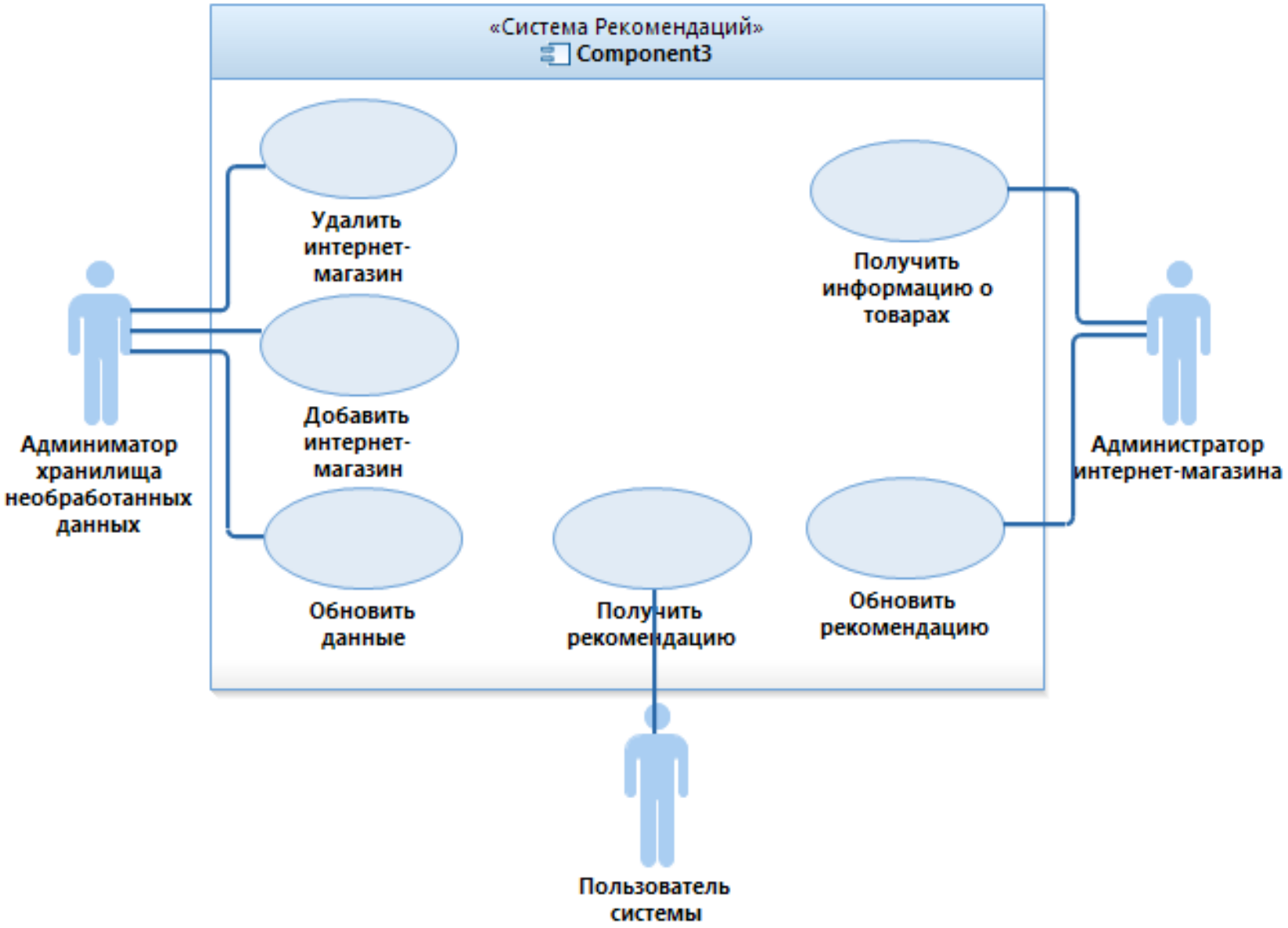



Modeling - Miscellaneous Models/Blank Use Case Package.emx - Rational® Software Architect
File Edit Navigate Search Project Run Modeling Window Help
Model
Modeling
Project Explorer
hhh
Diagrams
Models
Miscellaneous Models
Diagrams
Models
Blank Use Case Package
Main
(UMLPrimitiveTypes)
Blank Use Case Package.emx
Main
Alerts and Action Items
There are no problems associated with this model.
General Information
This section describes general information about this model.
Name: Blank Use Case Package
Location: \\profile01\FRW7\kutyrevame\Desktop\workspace_maya\Miscellaneous Models\Blank Use Case Package.emx
Size: 3 431 bytes
Last modified: 2 июня 2013 г. 18:56:11
Editable: true
Documentation
This section lists the documentation for this model.
Overview Details Diagrams References Fragments
Layers
Outline
Layers are not available.
Properties
Advanced
Property
Value
Info
derived false
editable true
last modified 2 июня 2013 г. 18:56:11
linked false
location \\profile01\FRW7\kutyrevame\Desktop\workspace_maya\Miscellan...
Свернуть все окна


Modeling - Blank Use Case Package::Main - Rational® Software Architect
File Edit Diagram Navigate Search Project Run Modeling Window Help
Model
Modeling
Project Explorer
hhh
Diagrams
Models
Miscellaneous Models
Diagrams
Models
Blank Use Case Package
Main
(UMLPrimitiveTypes)
Blank Use Case Package.emx
Main
Palette
Create
Explore
Use Case
Package
Use Case
Actor
Subsystem
Include
Association
Geometric Shapes
Sketch Shapes
Layers
Outline
This diagram does not have any layers added. Click here to add a new layer to this diagram.
Properties
<Model> Blank Use Case Package
General
Profiles
Stereotypes
Documentation
Constraints
Capabilities
Name:
Blank Use Case Package
Visibility:
Public
Private
Protected
Package
Viewpoint:
Blank Use Case Package

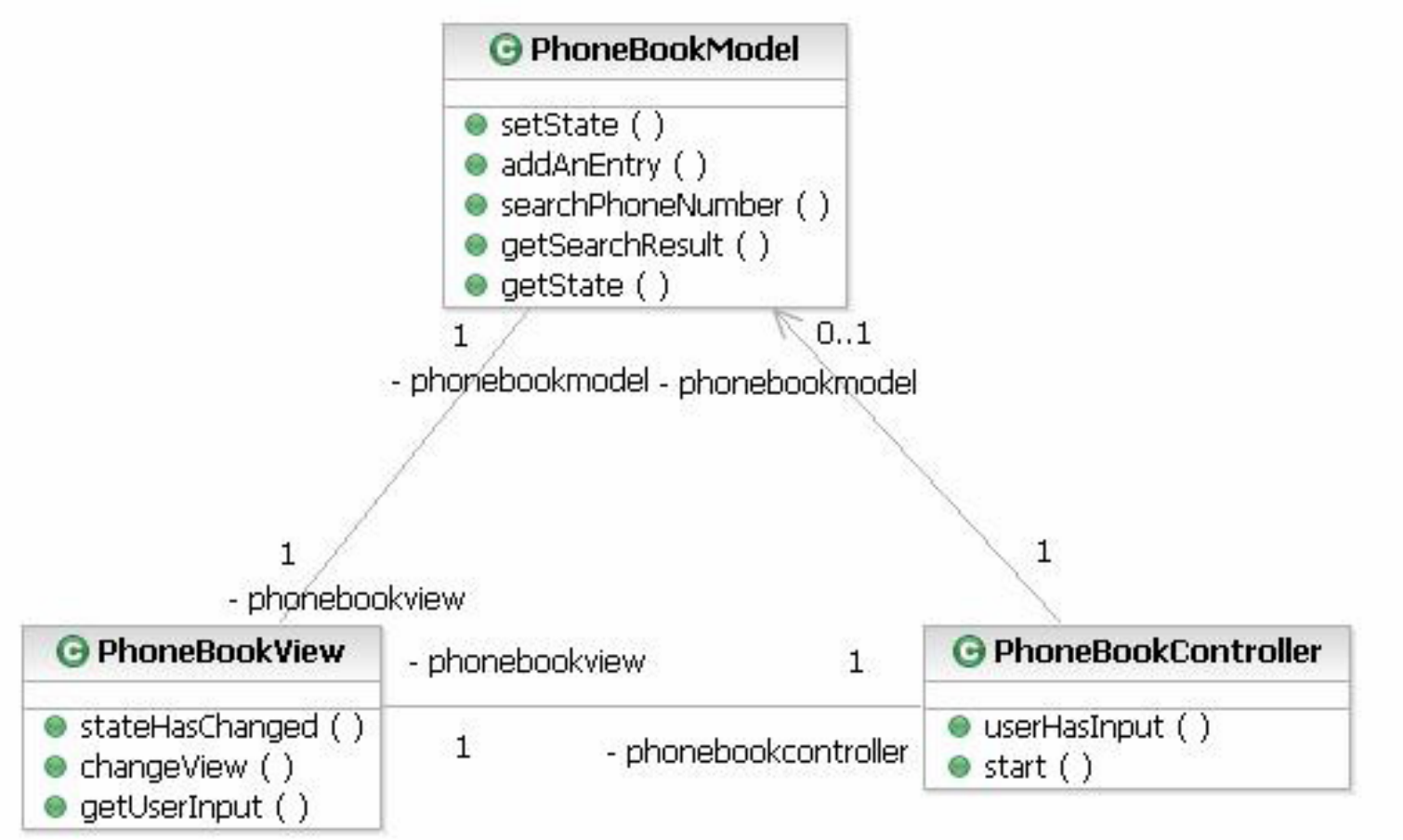
Что для необходимо знать для создания диаграммы use case?

- Актеры (существительное)
- Что они могут делать? (глагол)

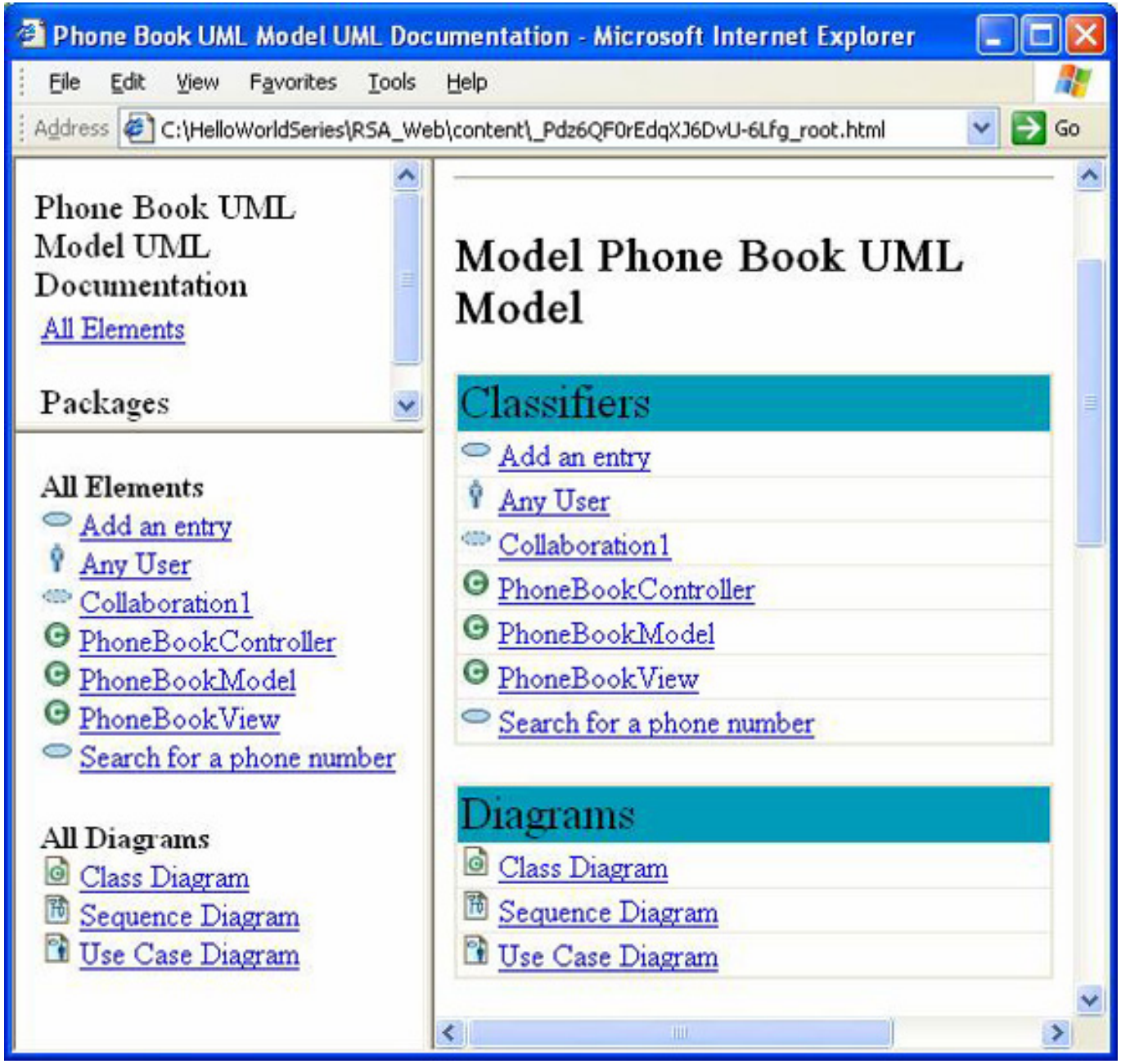

«Система Рекомендаций»
Component3
Удалить интернет-магазин
Добавить интернет-магазин
Обновить данные
Получить рекомендацию
Получить информацию о товарах
Обновить рекомендацию
Админиматор хранилища необработанных данных
Администратор интернет-магазина
Пользователь системы

- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- построение диаграммы последовательностей;
- опубликование проекта;
- непосредственное преобразование UML в Java.

Вот такой результат



- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- опубликование проекта;
- непосредственное преобразование UML в Java.

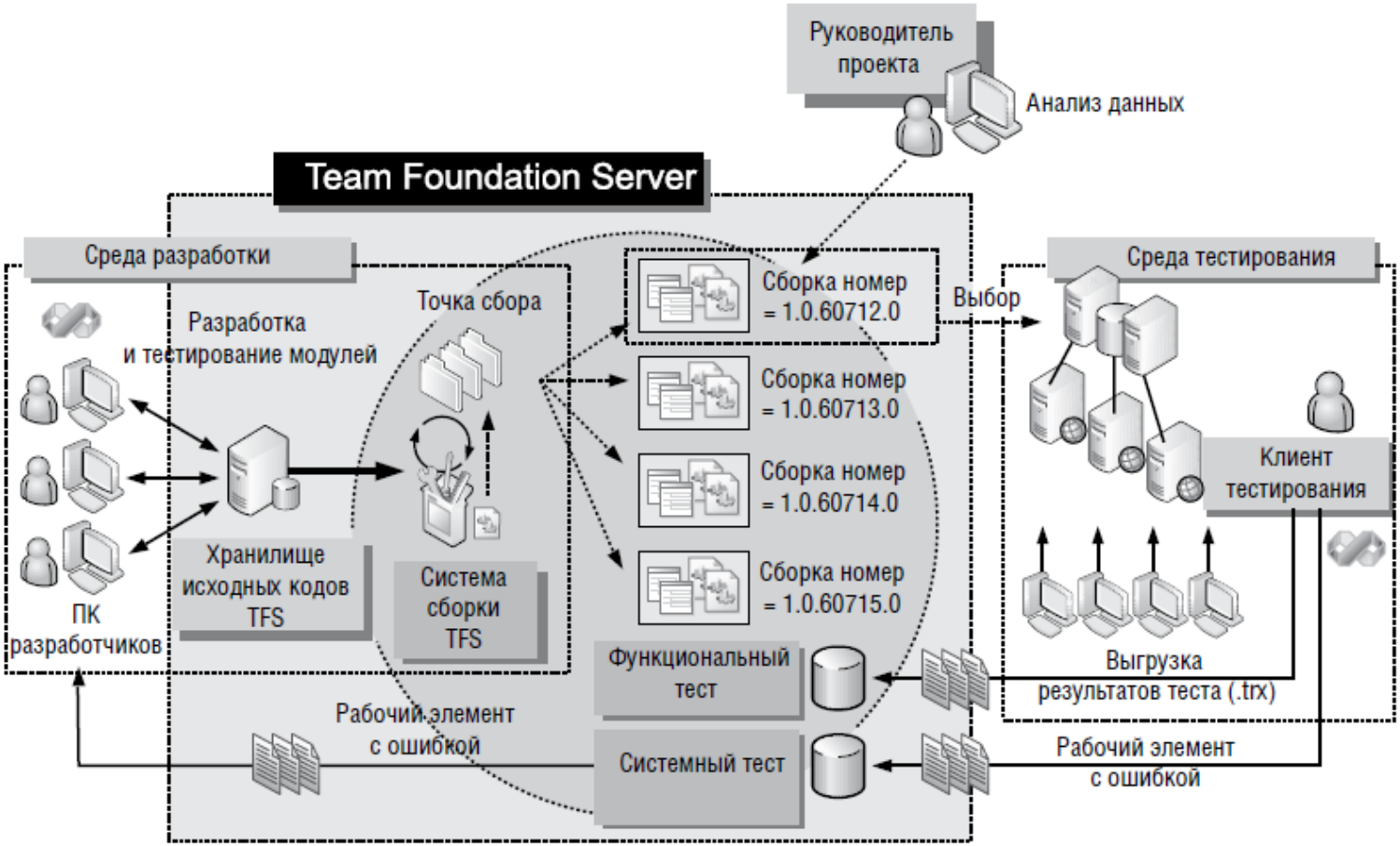

Phone Book UML Model UML Documentation - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address C:\HelloWorldSeries\RSA_Web\content_Pdz6QF0rEdqXJ6DvU-6Lfg_root.html
Go
Phone Book UML Model UML Documentation
All Elements
Packages
All Elements
Add an entry
Any User
Collaboration1
PhoneBookController
PhoneBookModel
PhoneBookView
Search for a phone number
All Diagrams
Class Diagram
Sequence Diagram
Use Case Diagram
Model Phone Book UML Model
Classifiers
Add an entry
Any User
Collaboration1
PhoneBookController
PhoneBookModel
PhoneBookView
Search for a phone number
Diagrams
Class Diagram
Sequence Diagram
Use Case Diagram

- создание UML проекта;
- построение use-case диаграммы;
- построение диаграммы классов;
- опубликование проекта;
- непосредственное преобразование UML в Java.

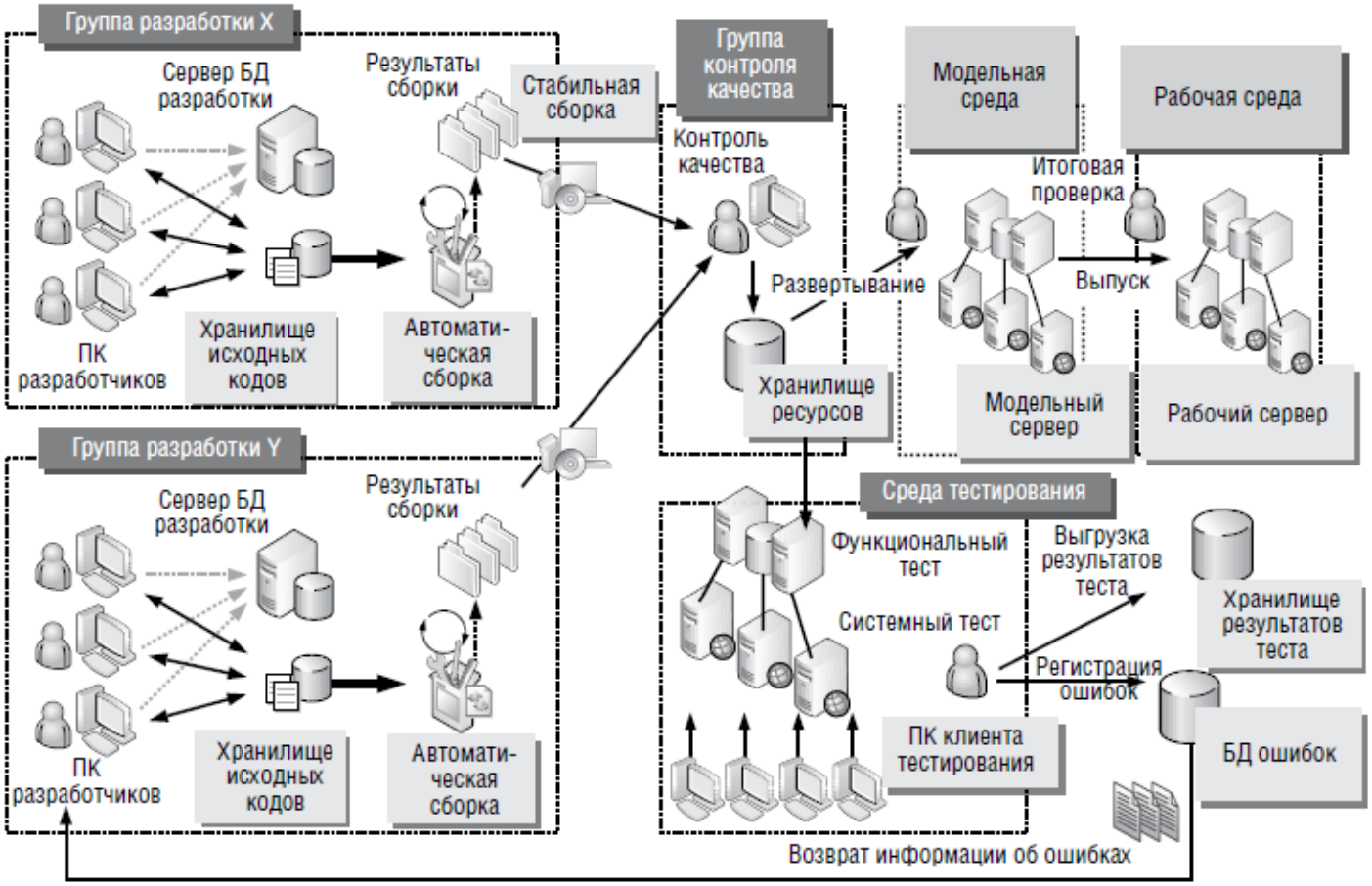
```
public class PhoneBookController {
    /**
     * Comment for <code>phonebookmodel</code>
     * @generated "UML to Java (com.ibm.xtools.transform.uml2
     */
    private PhoneBookModel phonebookmodel;

    /**
     * Comment for <code>phonebookview</code>
     * @generated "UML to Java (com.ibm.xtools.transform.uml2
     */
    private PhoneBookView phonebookview;

    /**
     * @generated "UML to Java (com.ibm.xtools.transform.uml2
     */
    public void userHasInput() {
        // TODO Auto-generated method stub
    }

    /**
     * @generated "UML to Java (com.ibm.xtools.transform.uml2
     */
    public void start() {
        // TODO Auto-generated method stub
    }
}
```

Системы поддержки коллективность разработки ПО (Microsoft Team Foundation Server)

Логический документооборот



Логическая организация работы в группах разработки



Физическая среда