МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ВЫСОКИХ ТЕХНОЛОГИЙ И ПЬЕЗОТЕХНИКИ

Кафедра прикладной информатики и инноватики

КУРСОВАЯ РАБОТА

по дисциплине

«Архитектура информационных систем»

на тему

«Сервис для поиска и покупки персонального компьютера»

Выполнил

студент 4 курса 3 группы

Пушенко Дмитрий Сергеевич

Принял

профессор, д.т.н.

Жмайлов Борис Борисович

Ростов-на-Дону

2022

1. Техническое задание

Предметная область курсовой работы связана с пользователями и информационными ресурсами, связанными с составляющими Персонального Компьютера (ПК). Проект должен представлять из себя библиотеку в формате jar архива.

Необходимо создать сервис поиска взаимосовместимых комплектующих ПК. Проект должен представлять собой Java приложение, предоставляющее API (application programming interface).

Предусмотреть в системе базовые операции: создание, редактирование, удаление и получение основных сущностей, таких как компьютер и его комплектующие (материнская плата, центральный процессор и оперативная память).

1. UML моделирование
   1. Диаграмма вариантов использования

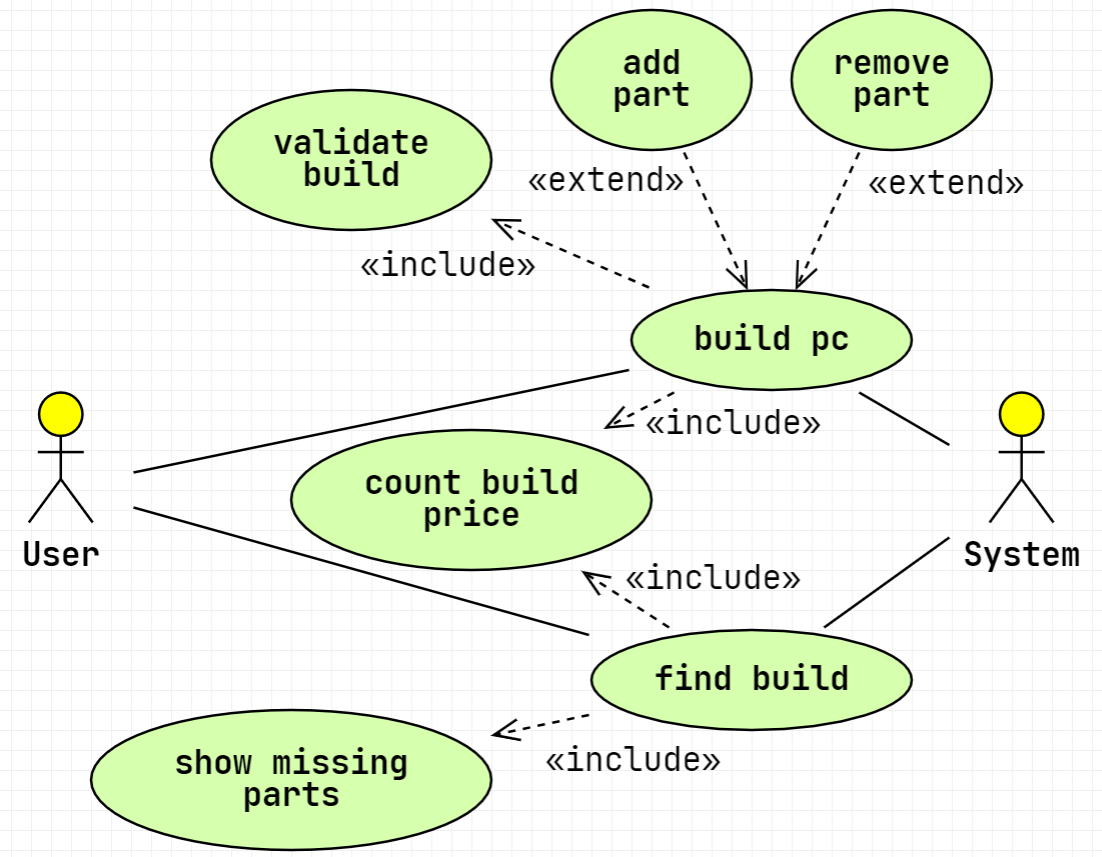


Схема 1. Диаграмма вариантов использования

Таблица 1. Детализация диаграммы вариантов использования.

|  |  |
| --- | --- |
| **Вариант использования** | **Описание** |
| Count build price | Посчитать стоимость конечной сборки.  **Входные данные:** long orderId;  **Выходные данные:** double cost. |
| **Build PC** | Собрать конфигурацию ПК.  **Входные данные:** long orderId, String action, long partId;  **Выходные данные:** Optional<Order> order;  **Включающие методы** «Validate build» и «Count build price» получают параметры из сигнатуры родительского метода.  **Расширяющие методы** «Add part» и «Remove part» получают параметры из сигнатуры родительского метода и выполняются при action.equals(«add») и action.equals(«remove») соответственно. |
| Validate build | Проверка комплектующих на совместимость и нехватку необходимых.  Возвращает false если установлены несовместимые, или не выбраны необходимые комплектующие.  **Входные данные:** long orderId;  **Выходные данные** bool isCorrect. |
| Add part | Добавить деталь в сборку.  **Входные данные:** long orderId, long partId;  **Выходные данные:** Optional<Order> order. |
| Remove part | Удалить деталь из сборки.  **Входные данные:** long orderId, long partId;  **Выходные данные:** Optional<Order> order. |
| **Find build** | Найти сохранённую конфигурацию ПК.  **Входные данные:** long orderId;  **Выходные данные:** Optional<Order> order;  **Включающие методы** «Show missing parts» и «Count build price» получают параметры из сигнатуры родительского метода. |
| Show missing parts | Поиск недостающих деталей конфигурации.  **Входные данные:** long orderId;  **Выходные данные:** List<Part> parts. |

* 1. Диаграмма классов

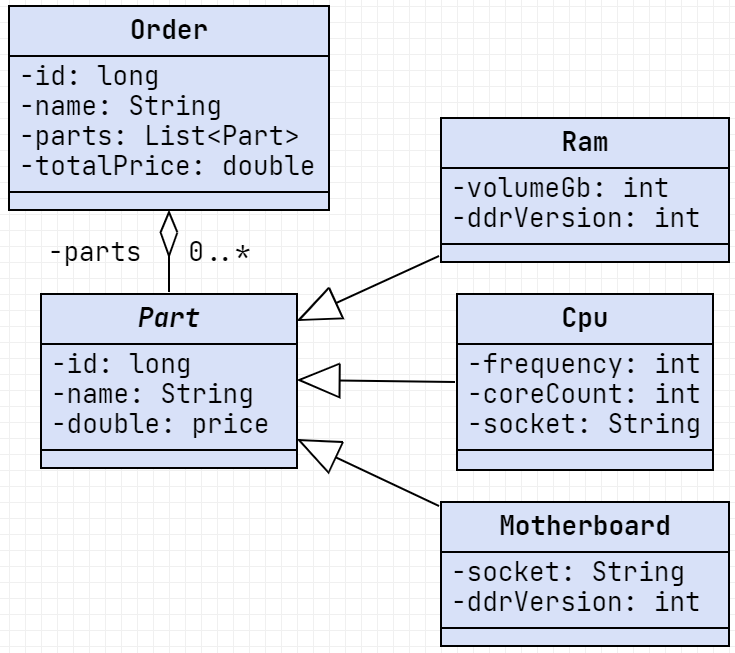


Схема 2. Диаграмма классов

* 1. Диаграммы активностей

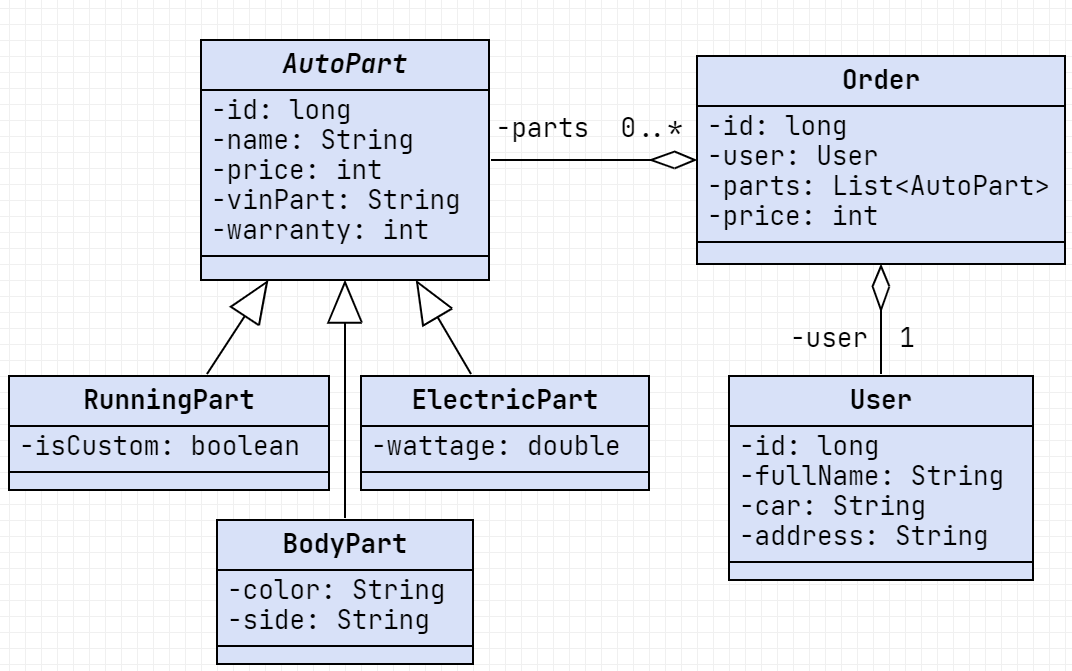


Схема 3. Диаграмма классов

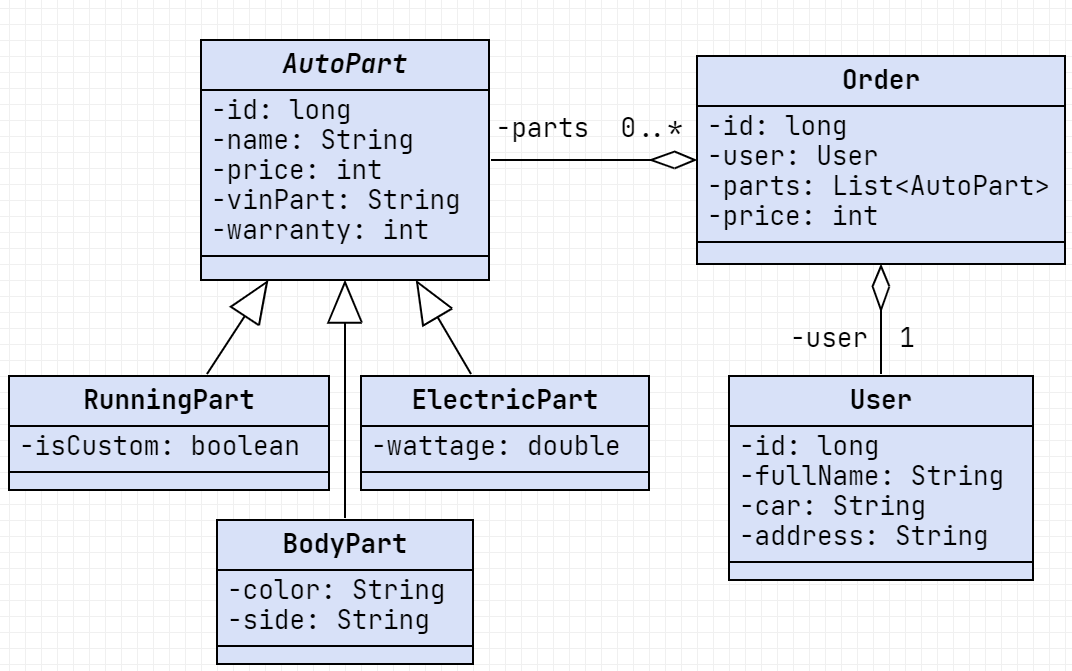


Схема 4. Диаграмма классов

* 1. Диаграмма компонентов

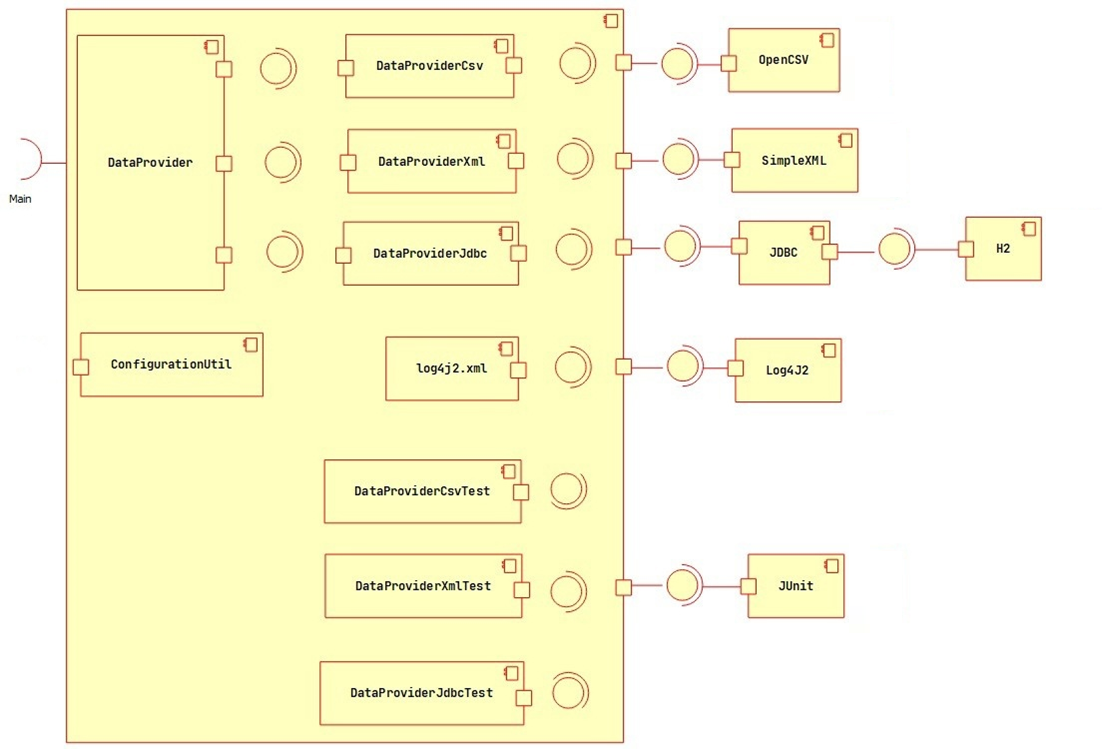


Схема 5. Диаграмма компонентов

1. Сценарии тестирования и результаты тестов

Таблица 2. Сценарии и результаты тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| **Тестовый метод** | **Тип сценария** | **Тестируемый метод** | **Что контролируется?** |
| countBuildPricePos | позитивный | DataProvider.countBuildPrice | Assertions.assertEquals(expectedPrice, actualPrice) |
| countBuildPriceNeg | негативный | DataProvider. countBuildPrice | Assertions.assertEquals(expectedPrice, actualPrice) |
| buildPcPos | позитивный | DataProvider.buildPc | Assertions.assertEquals(expectedOrder, actualOrder) |
| buildPcNeg | негативный | DataProvider.buildPc | Assertions.assertNotEquals(expectedOrder, actualOrder) |
| addPartPos | позитивный | DataProvider.addPart | Assertions.assertEquals(expectedOrder, actualOrder) |
| addPartNeg | негативный | DataProvider.addPart | Assertions.assertNotEquals(expectedOrder, actualOrder) |
| removePartPos | позитивный | DataProvider.removePart | Assertions.assertEquals(expectedOrder, actualOrder) |
| removePartNeg | негативный | DataProvider.removePart | Assertions.assertEquals(expectedOrder, actualOrder) |
| validateBuildPos | позитивный | DataProvider.validateBuild | Assertions.assertTrue(dp.validateBuild(o1.getId())) |
| validateBuildNeg | негативный | DataProvider.validateBuild | Assertions.assertFalse(dp.validateBuild(o2.getId())) |
| findBuildPos | позитивный | DataProvider.findBuild | Assertions.assertEquals(expectedOrder, actualOrder) |
| findBuildNeg | негативный | DataProvider.findBuild | Assertions.assertEquals(expectedOrder, actualOrder) |
| showMissingPartsPos | позитивный | DataProvider.showMissingParts | Assertions.assertEquals(expectedParts, actualParts) |
| showMissingPartsNeg | негативный | DataProvider.showMissingParts | Assertions.assertEquals(expectedParts, actualParts) |