

PROJECT “SHOP”

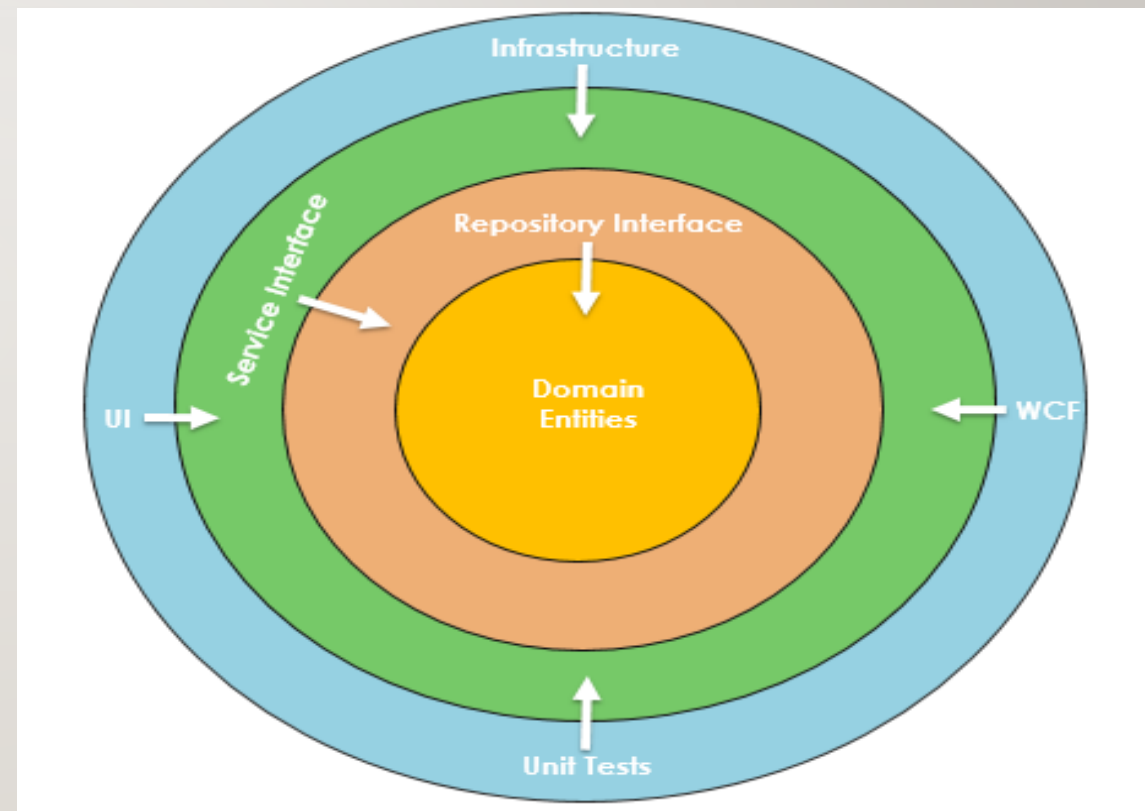
MENTOR: ANDREY POMELOV

РАЗРАБОТКА ПРОЕКТА «МАГАЗИН»

- Принципы проектирования приложений. «Чистая архитектура».
- Задача проекта.
- Описание сущностей проекта.
- Описание слоёв проекта.
- Тестирование созданного приложения.

«ЧИСТАЯ АРХИТЕКТУРА»

«**Чистая архитектура**» — это философия дизайна программного обеспечения, которая разделяет компоненты на определенные уровни.



ЗАДАЧА ПРОЕКТА

Разработать проект «Магазин», руководствуясь принципами «чистой архитектуры».
Доменной (предметной) областью приложения являются продукты, покупатели, а также их корзины, в которые покупатели могут складывать выбранные продукты.

ОПИСАНИЕ СУЩНОСТЕЙ ПРОЕКТА

ПРОДУКТ.

Продукт имеет следующие свойства:

- Целочисленный уникальный идентификатор
- Логический параметр, указывающий, активен ли продукт
- Строковое наименование
- Цена (может иметь дробные значения)

ОПИСАНИЕ СУЩНОСТЕЙ ПРОЕКТА

ПОКУПАТЕЛЬ.

Покупатель имеет следующие свойства:

- Целочисленный уникальный идентификатор
- Логический параметр, указывающий, активен ли покупатель
- Строковое имя
- Объект корзины

ОПИСАНИЕ СУЩНОСТЕЙ ПРОЕКТА

КОРЗИНА.

Корзина имеет следующие свойства:

- Целочисленный уникальный идентификатор
- Список товаров, лежащий в этой корзине

ФУНКЦИОНАЛ КОРЗИНЫ.

- Добавить продукт в корзину (если активный)
- Получение всех продуктов, находящихся в корзине (активных)
- Удалить продукт из корзины по его идентификатору
- Полная очистка корзины (удаление всех продуктов)
- Получение общей стоимости корзины (активных продуктов)
- Получение средней стоимости товара в корзине (из активных продуктов)

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Первый слой. Домен.

Содержит все классы, описывающие сущности проекта.

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Второй слой. Репозитории.

Проект должен содержать репозиторий продуктов и репозиторий покупателей.

Репозиторий – это класс, объект которого служит для доступа к базе данных. Физическое хранение объектов сущностей осуществляется в самой БД.

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Функционал репозитория.

- Сохранить новый объект в базе данных. При этом каждому новому объекту автоматически должен присваиваться автоинкрементируемый уникальный идентификатор (обеспечивается функционалом базы данных).
- Вернуть список всех объектов.
- Вернуть один объект по его идентификатору.
- Изменить один объект по его идентификатору.
- Удаление одного объекта из базы данных по его идентификатору. При этом физического удаления происходить не должно, вместо этого логический параметр объекта просто должен быть выставлен в ложное значение. Такой функционал необходим для возможности восстановления удалённых объектов.

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Третий слой. Сервисы.

Проект должен содержать сервис продуктов и сервис покупателей.

Сервис – это класс, объект которого обеспечивает всю бизнес-логику приложения, касающуюся сущностей, с которыми работает приложение.

Для сохранения, получения, модификации и удаления сущностей из базы данных сервис обращается к соответствующему репозиторию.

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Функционал сервиса продуктов.

- Сохранить продукт в базе данных (при сохранении продукт автоматически считается активным).
- Вернуть все продукты из базы данных (активные).
- Вернуть один продукт из базы данных по его идентификатору (если он активен).
- Изменить один продукт в базе данных по его идентификатору.
- Удалить продукт из базы данных по его идентификатору.
- Удалить продукт из базы данных по его наименованию.
- Восстановить удалённый продукт в базе данных по его идентификатору.
- Вернуть общее количество продуктов в базе данных (активных).
- Вернуть суммарную стоимость всех продуктов в базе данных (активных).
- Вернуть среднюю стоимость продукта в базе данных (из активных).

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Функционал сервиса покупателей.

- Сохранить покупателя в базе данных (при сохранении покупатель автоматически считается активным).
- Вернуть всех покупателей из базы данных (активных).
- Вернуть одного покупателя из базы данных по его идентификатору (если он активен).
- Изменить одного покупателя в базе данных по его идентификатору.
- Удалить покупателя из базы данных по его идентификатору.
- Удалить покупателя из базы данных по его имени.
- Восстановить удалённого покупателя в базе данных по его идентификатору.
- Вернуть общее количество покупателей в базе данных (активных).
- Вернуть стоимость корзины покупателя по его идентификатору (если он активен).
- Вернуть среднюю стоимость продукта в корзине покупателя по его идентификатору (если он активен)
- Добавить товар в корзину покупателя по их идентификаторам (если оба активны)
- Удалить товар из корзины покупателя по их идентификаторам
- Полностью очистить корзину покупателя по его идентификатору (если он активен)

ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Четвёртый слой. Контроллеры.

Проект должен содержать контроллер продуктов и контроллер покупателей.

Контроллер – это класс, объект которого принимает запросы, поступившие приложению от клиента, решает, что с ними делать, предпринимает необходимые действия и возвращает ответ клиенту. Например, если от клиента поступил запрос на получение списка всех продуктов, контроллер обращается к сервису, получает список всех продуктов и отправляет его клиенту.

Для выполнения всех необходимых действий, связанных с запросами клиентов, контроллер обращается к сервису.



ОПИСАНИЕ СЛОЁВ ПРОЕКТА

Функционал контроллеров.

Контроллеры продуктов и покупателей должны иметь методы, реализующие все возможности, предоставляемые соответствующими сервисами.

ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ

Тестирование приложения можно осуществить при помощи любого стороннего приложения, которое позволяет отправлять http-запросы. Например, Postman.