

# Лабораторная 5

## Отрабатываемый материал

Многослойные архитектуры, паттерны

## Цель

Проверить освоение студентом многослойных архитектур

## Задание

Реализовать систему банкомата

## Функциональные требования

- создание счета
- просмотр баланса счета
- снятие денег со счета
- пополнение счета
- просмотр истории операций

## Не функциональные требования

- интерактивный консольный интерфейс
- возможность выбора режима работы (пользователь, администратор)
  - при выборе пользователя должны быть запрошены данные счета (номер, пин)
  - при выборе администратора должен быть запрошен системный пароль
    - при некорректном вводе пароля - система прекращает работу
- системный пароль должен быть параметризуем
- при попытке выполнения некорректных операций, должны выводиться сообщения об ошибке
- данные должны быть персистентно сохранены в базе данных (PostgreSQL)
- использование каких-либо ORM библиотек - запрещено
- приложение должно иметь гексагональную архитектуру
  - опционально: можно реализовать луковую архитектуру с богатой доменной моделью.

## Test cases

- снятие денег со счёта
  - при достаточном балансе проверить что сохраняется счёт с корректно обновлённым балансом
  - при недостаточном балансе сервис должен вернуть ошибку

- пополнение счёта
  - проверить что сохраняется счёт с корректно обновлённым балансом

данные тесты должны проверять бизнес логику, они не должны как-либо зависеть от базы данных или консольного представления.

в данных тестах необходимо использовать моки репозиториев.

## Ссылки

<https://spectreconsole.net>

<https://www.npgsql.org>

<https://www.nuget.org/packages/NSubstitute>