

Въведение в технологиите

Frontend

В MVP-то ще започнем с **React** за web версия, а Flutter ще добавим в Итерация 3

Backend

Ще работим с **Node.js** за MVP, защото:

- По-лесна интеграция с React
- Отлична performance
- По-бързо време за development

Database

Работим с **PostgreSQL**, защото нашите данни са силно структурирани:

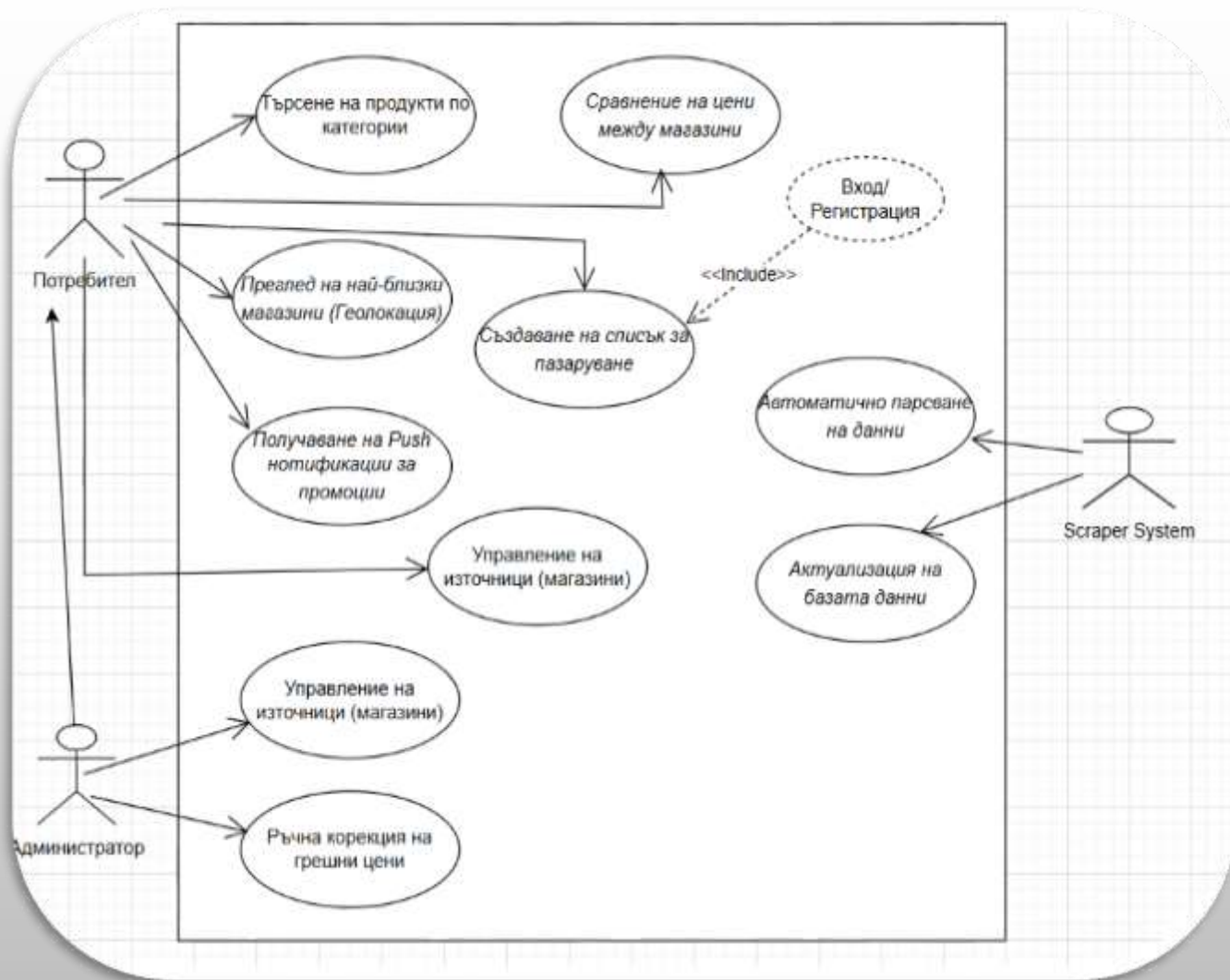
- Продукти с категории
- Цени свързани с магазини и дати
- Потребителски списъци

Имаме нужда от **ACID транзакции** за точност на данните.

Data Scraping

Scraper-ът за Kaufland ще е:

- Отваря категория 'Мляко и млечни продукти'
- Извлича име на продукт, цена, единица
- Съхранява в базата данни
- Повтаря процеса на всеки 24 часа

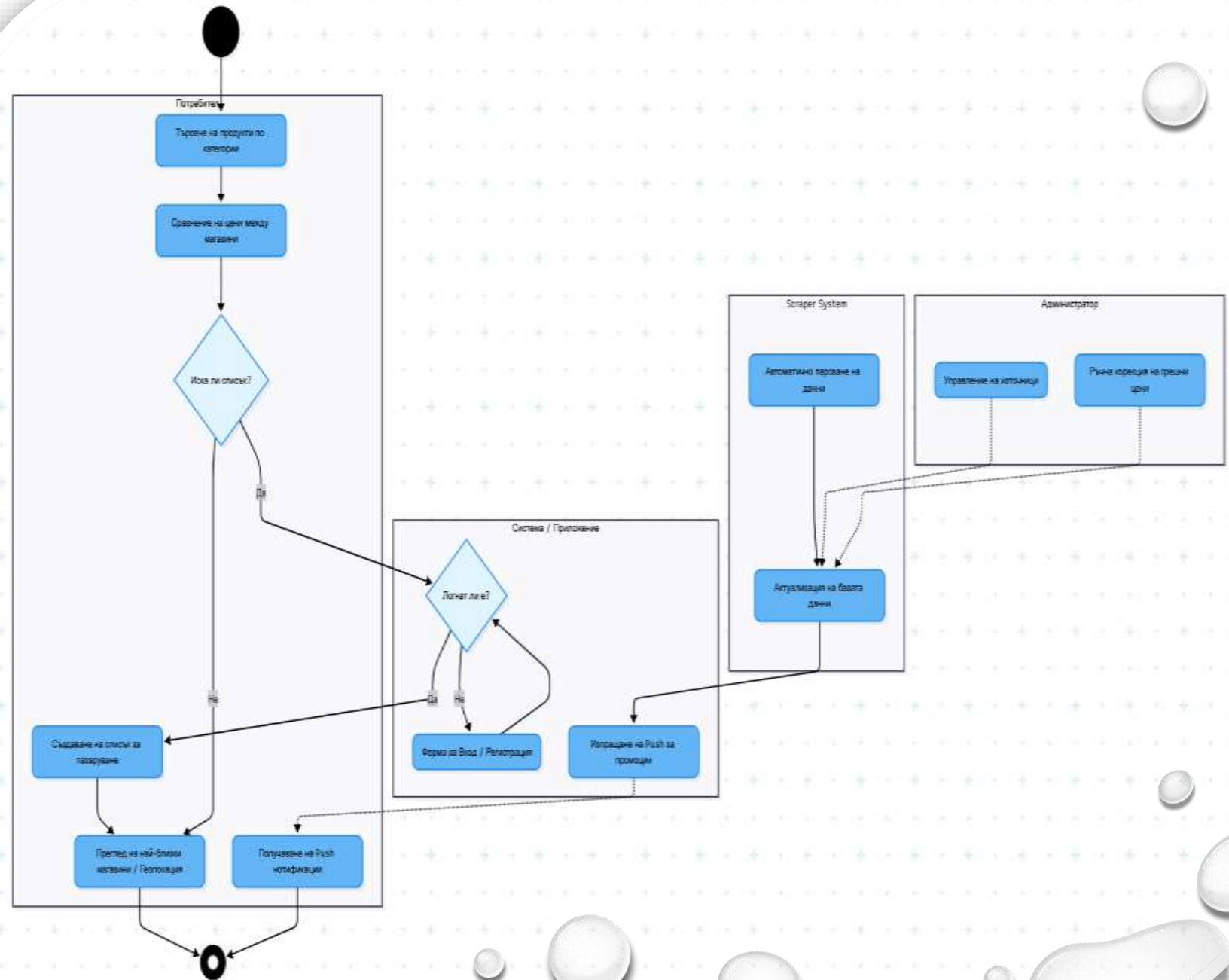


USE CASE диаграма

UML ДИАГРАМАТА ПОКАЗВА
ОСНОВНИТЕ ФУНКЦИОНАЛНОСТИ НА
НАШИЯ ПРОЕКТ, КОЙТО
РАЗРАБОТВАМЕ.

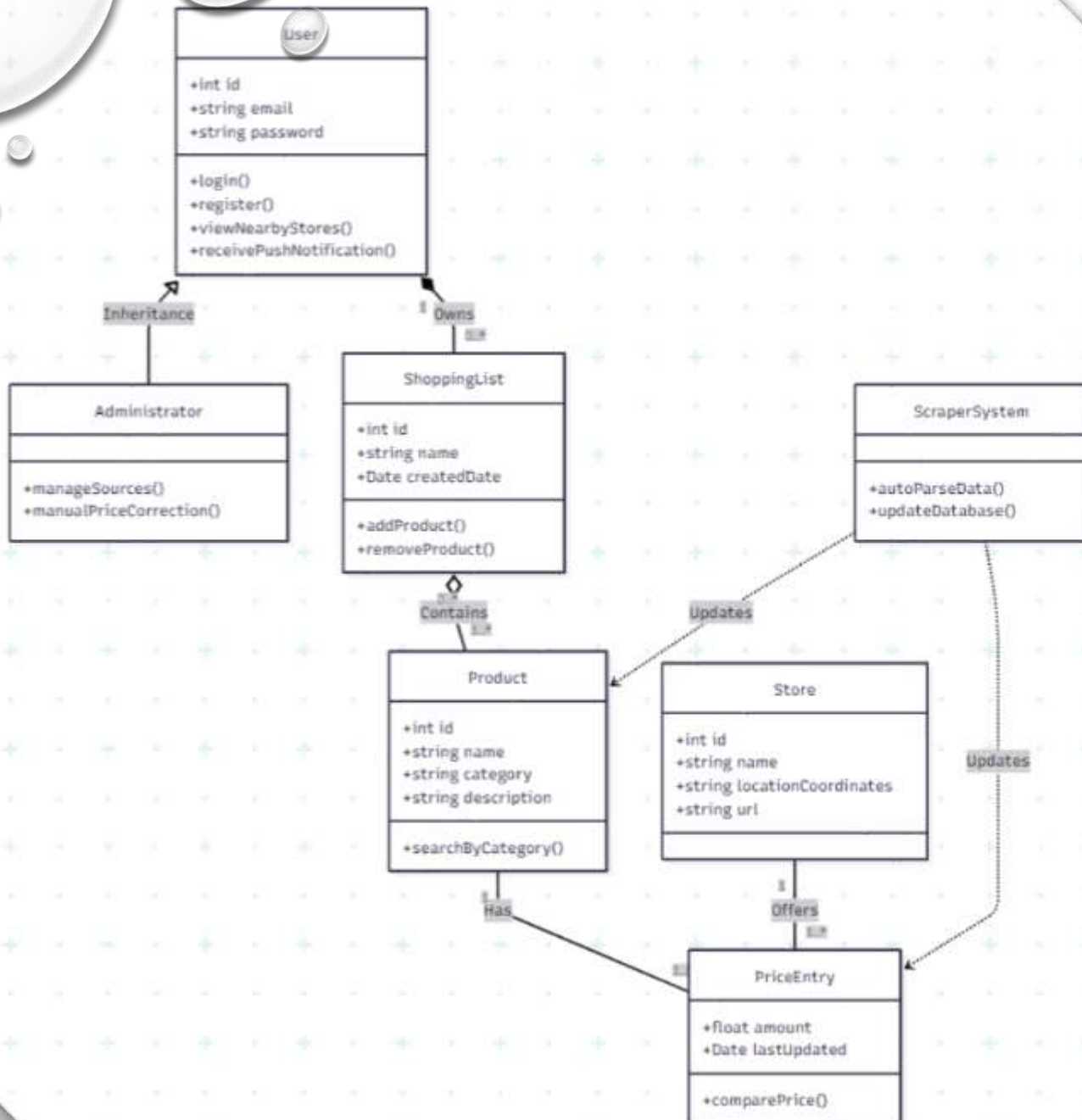
Activity диаграмата

- **ACTIVITY ДИАГРАМАТА** ОПИСВА ПОСЛЕДОВАТЕЛНОСТТА ОТ ДЕЙСТВИЯ И ЛОГИЧЕСКИТЕ РАЗКЛОНЕНИЯ В ЕДИН КОНКРЕТЕН ПРОЦЕС, КАТО ПОКАЗВА СЪПКА ПО СЪПКА КАК СЕ ДВИЖИ РАБОТНИЯТ ПОТОК В СИСТЕМАТА.



Class диаграмата

- **CLASS ДИАГРАМАТА** Е СТРУКТУРЕН ИНСТРУМЕНТ В UML, КОЙТО ОПИСВА СТАТИЧНИЯ ИЗГЛЕД НА СИСТЕМАТА, КАТО ПРЕДСТАВЯ НЕЙНИТЕ КЛАСОВЕ, ТЕХНИТЕ АТРИБУТИ, МЕТОДИ И ВРЪЗКИТЕ МЕЖДУ ТЯХ.



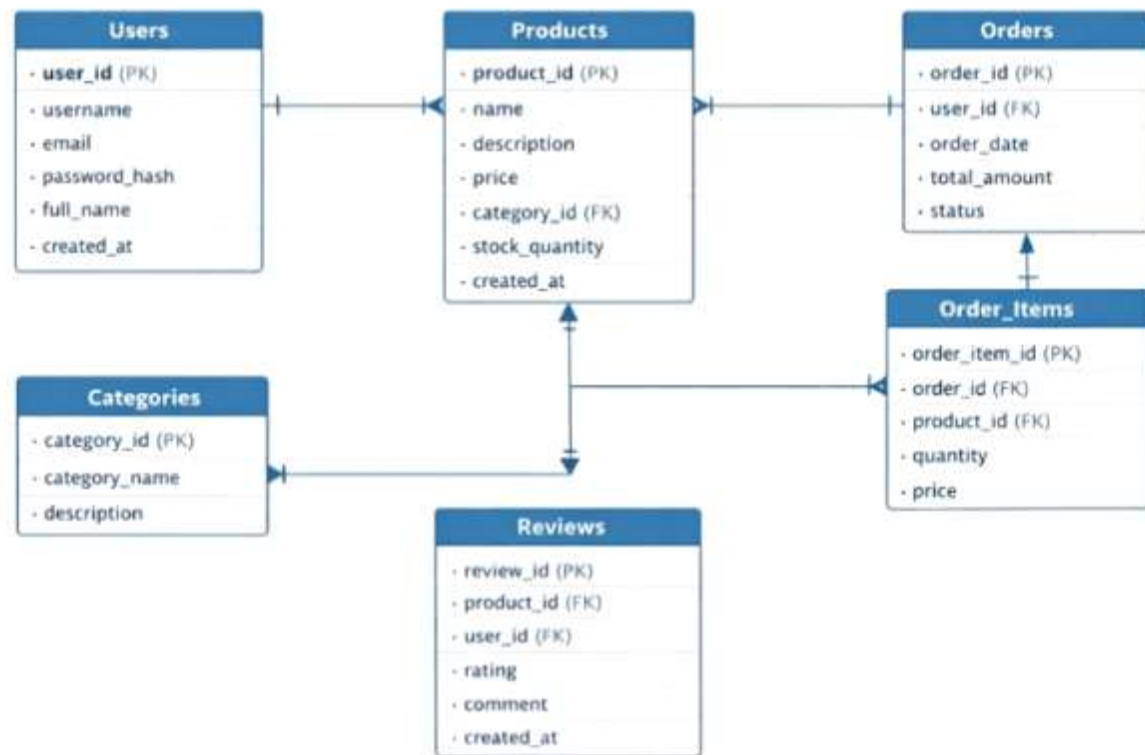


Схема на отделните Database обекти

- СХЕМАТА НА DATABASE ОБЕКТТЕ ОПИСВА ТАБЛИЦИТЕ, ТЕХНИТЕ АТРИБУТИ И ВРЪЗКИТЕ МЕЖДУ ТЯХ, НЕОБХОДИМИ ЗА СТРУКТУРИРАНО, НАДЕЖДНО И ЕФЕКТИВНО СЪХРАНЕНИЕ НА ДАННИТЕ В ПРИЛОЖЕНИЕТО.