

Система извлечения терминов

Сапожков Андрей ИУ7-63Б

Цель и задачи

Цель: разработка системы извлечения многокомпонентных терминов из параллельных текстов.

Задачи:

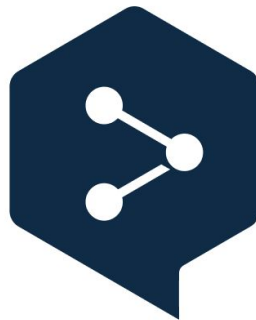
1. Провести анализ предметной области и формализовать задачу
2. Спроектировать БД и структуру ПО
3. Реализовать интерфейс для доступа к БД
4. Реализовать приложение для работы с БД

Анализ

Разрабатываемая система предназначена для создания терминологической БД и проведения исследований в области компьютерной лингвистики.

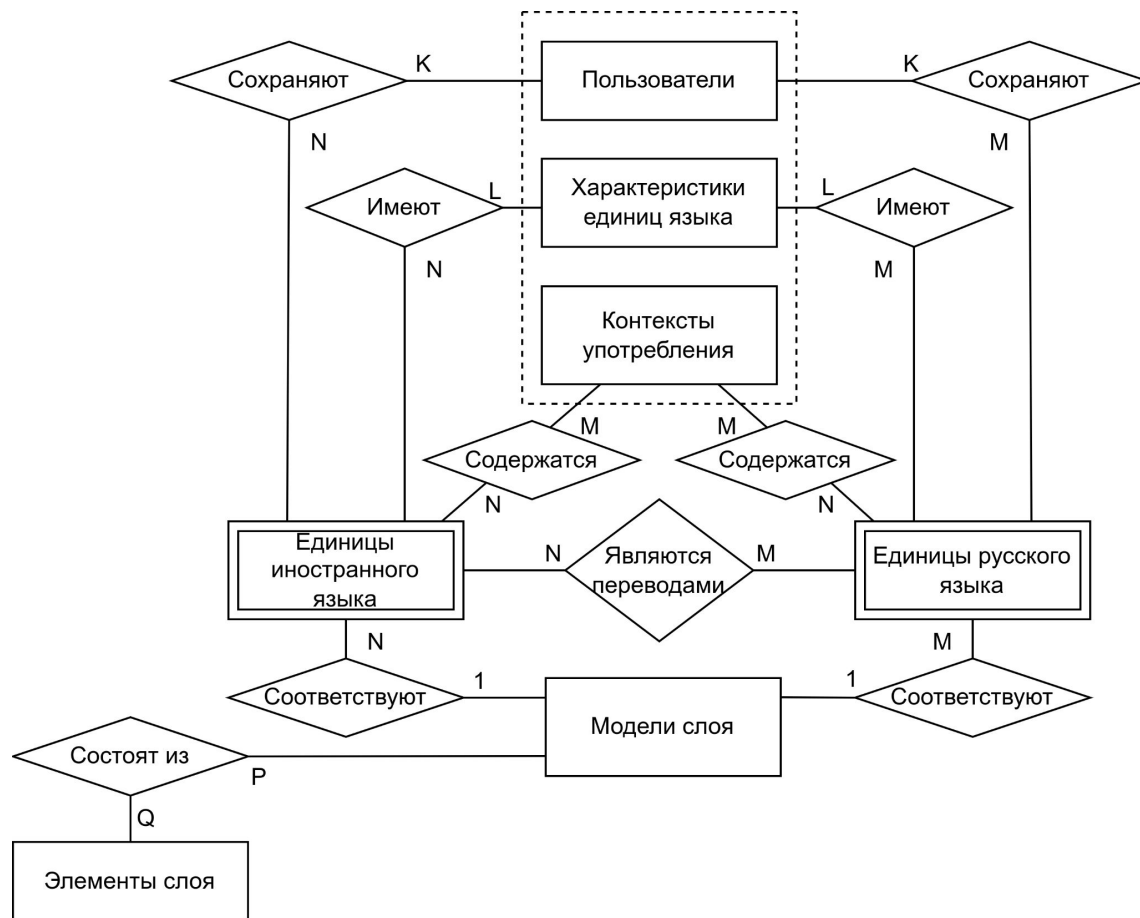
Отдалённые аналоги:

1. Переводчики (Google, Яндекс, DeepL)
2. Словари (Thesaurus)



Проектирование

Многослойная ER-модель



Структура проекта на Golang

Standard Go project layout:

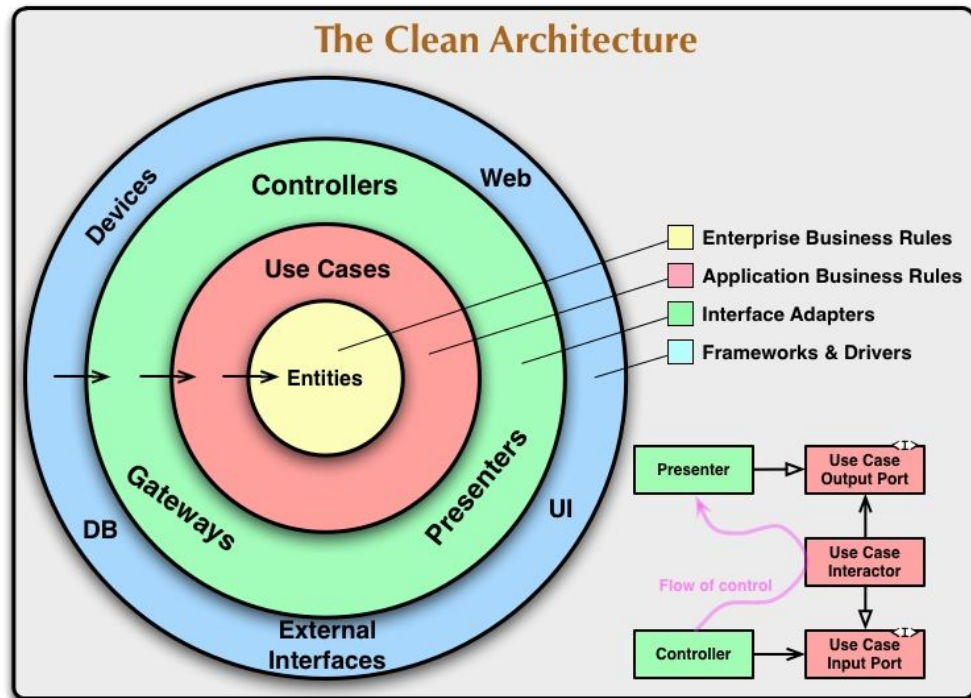
- **cmd** - код, из которого будут собираться бинарники;
- **internal** - внутренний код приложения и библиотек;
- **pkg** - код библиотек, которые могут использоваться в сторонних проектах;

<https://github.com/golang-standards/project-layout>

```
.  
├── Makefile  
├── api  
├── assets  
├── build  
├── cmd  
├── configs  
├── deployments  
├── docs  
├── examples  
├── githooks  
├── init  
├── internal  
├── pkg  
├── scripts  
├── test  
├── third_party  
├── tools  
├── vendor  
├── web  
└── website
```

Чистая архитектура

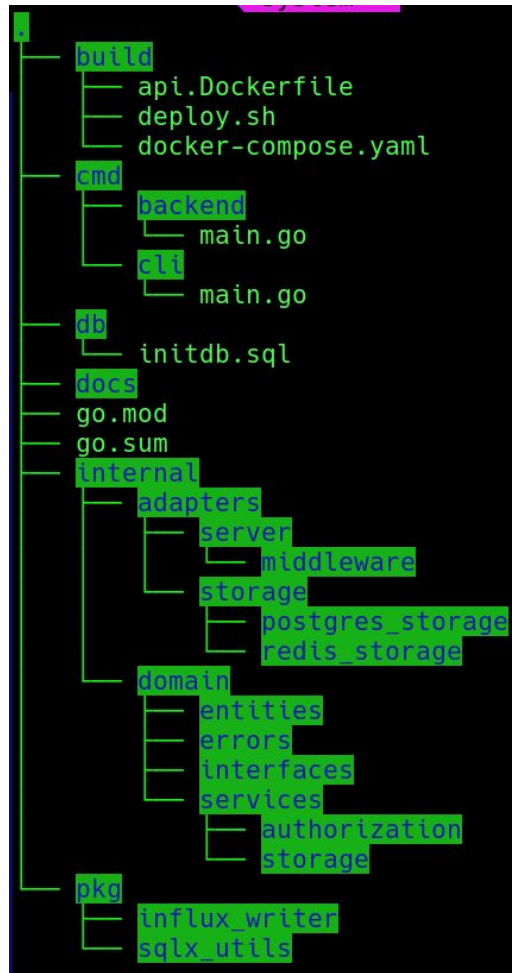
- Внутренние слои НЕ зависят от внешних
- Зависимости в исходном коде могут указывать только во внутрь
- Обращение к доменной области через интерфейсы (dependency injection)



Чистая архитектура в Golang

2 способа группировать код:

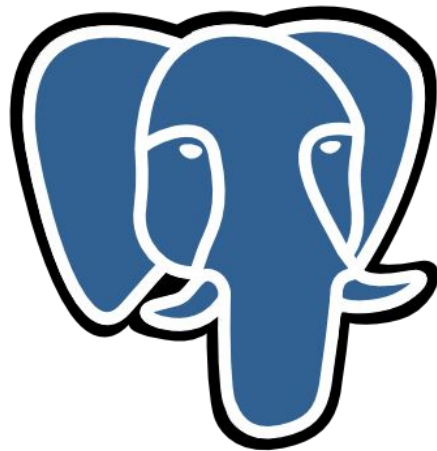
- + Вертикальные срезы
(делить по сценариям использования)
- + Горизонтальные слои
(делить по причинам для изменений)



База данных

PostgreSQL:

- + Многочисленные расширения
- + Объектно-реляционная модель
- + Open-source
- + Сертификация ФСТЭК
- + Имеется опыт использования
- + Имеется желание прочувствовать всю боль администрирования БД

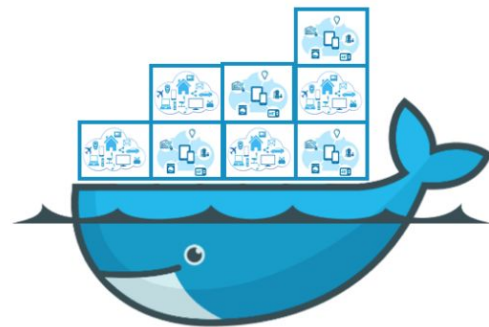


PostgreSQL

Логирование (1)

Писать логи в Docker контейнер:

- + Просто реализовать
- Небезопасно
- Сложно собирать информацию в один таймлайн



Писать логи в файл:

- + Просто реализовать
- Ещё более небезопасно
- Где хранить файлы логов?



Логирование (2)

Сервер логов:

- Необходимо настраивать СУБД
- + Вся ответственность за хранение лежит на СУБД
- + Единая точка сбора информации
- + СУБД временных рядов красиво располагает данные на таймлайне
- + InfluxDB имеет константную скорость записи
- + ЮВ подпишет курсовую по БД



Дальнейшие действия

- Сделать UI
- Внедрить кеш (Redis)
- Закрывать ППО
- Закрывать БД