Pet Shop

Project x

Team : Brothers in IT

Brothers in IT

Team members:

Bekmaganbetov Zhanbolat

Khamze Sultan Bibarys

Tolegen Assel

Contents of This Template

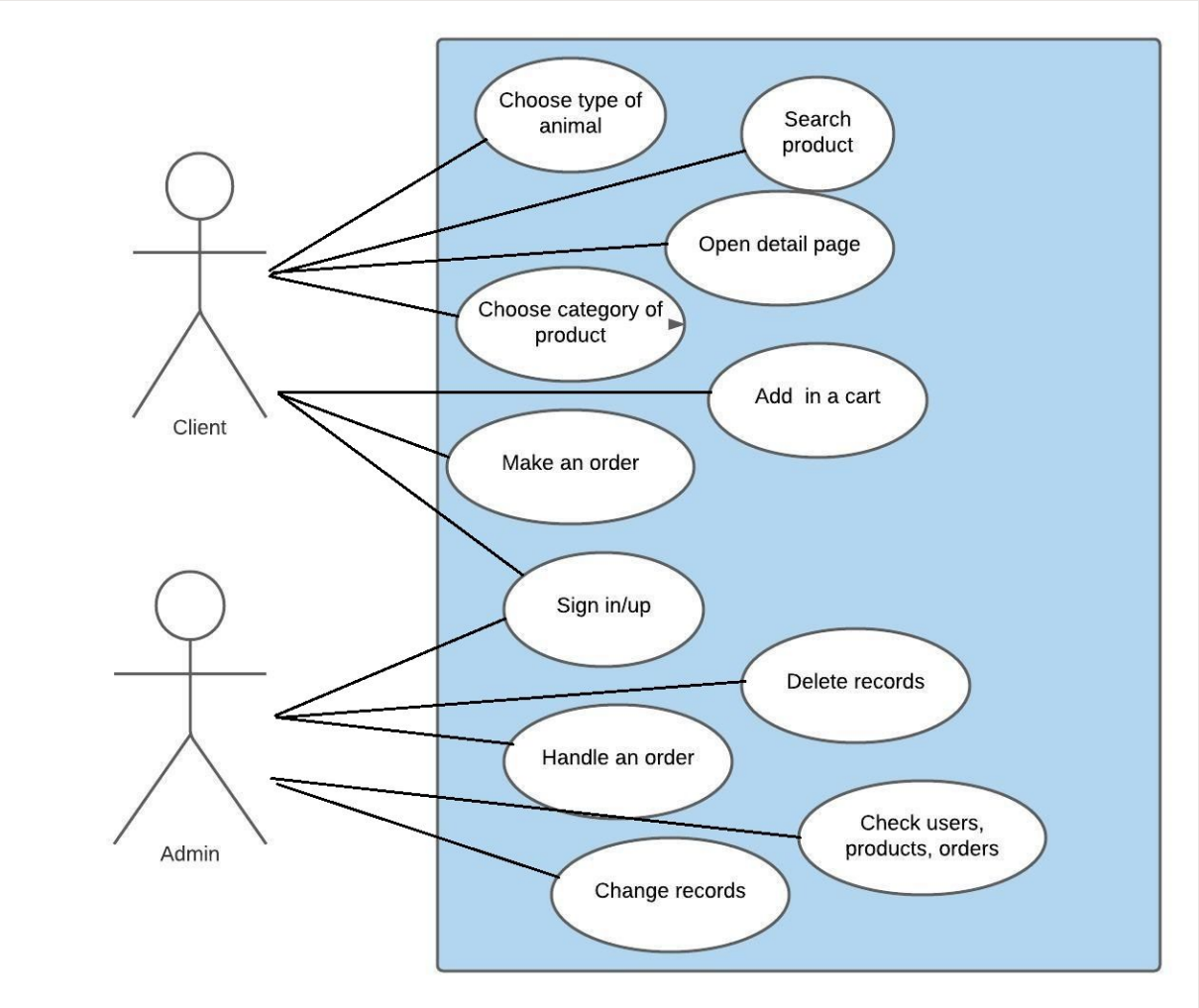
Here's what you'll find in this **Slidesgo** template:

1. Description
2. Use-case diagram
3. Data modeling
4. **Data sources**
5. Database structure:
    - Tables
    - Speed Performance
    - Queries
6. Experience and problems

Description!

During the time of Pandemie a lot of people today must to stay home and reduce all their contacts with other people. As the result people are staying either their pets or absolutely alone and order to minimize contacts with other , it’s become enough problematic and dangerous to visit such habitual places like pet shop , where we are always buying everything what is necessary for our pets , including medications, snacks , feed etc. This project can help people to find and purchase everything for their favourite little friends

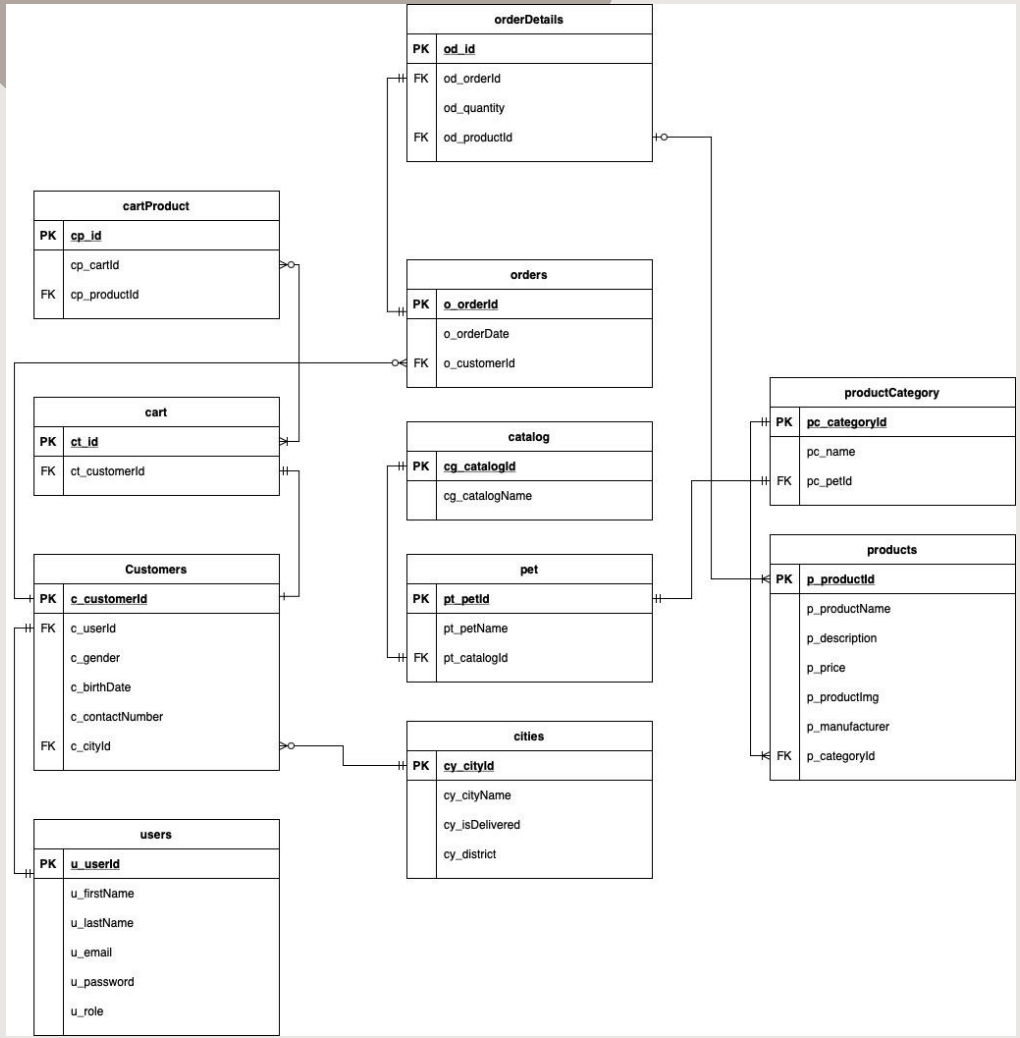
Description


Choose type of animal
Search product
Open detail page
Choose category of product
Add in a cart
Make an order
Sign in/up
Delete records
Handle an order
Check users, products, orders
Change records
Client
Admin

ERD diagram

Contains 11 tables:

- Users
- Customers
- Catalog
- ProductCategory
- Products
- Cities
- Cart
- CartProduct
- Pet
- Orders
- OrderDetails



TechStack

Frontend and Backend

TechStack

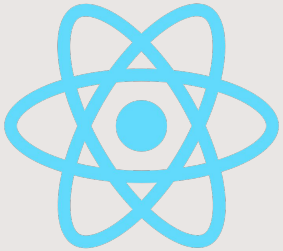
Frontend:

- ReactJS (React-bootstrap, React Router)
- Mobx

Backend:

- NodeJS
- ExpressJS
- PostgresSQL

TechStack



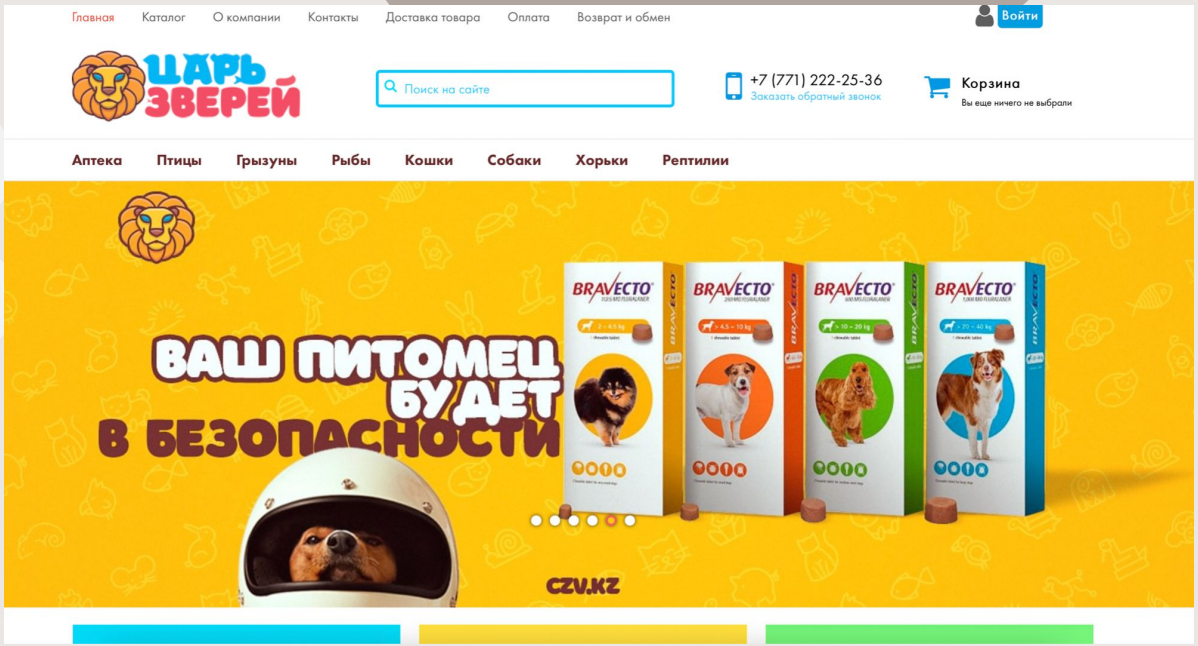
ReactJS

React JS is a JavaScript library used in web development to build interactive elements on websites.

NodeJs / ExpressJS

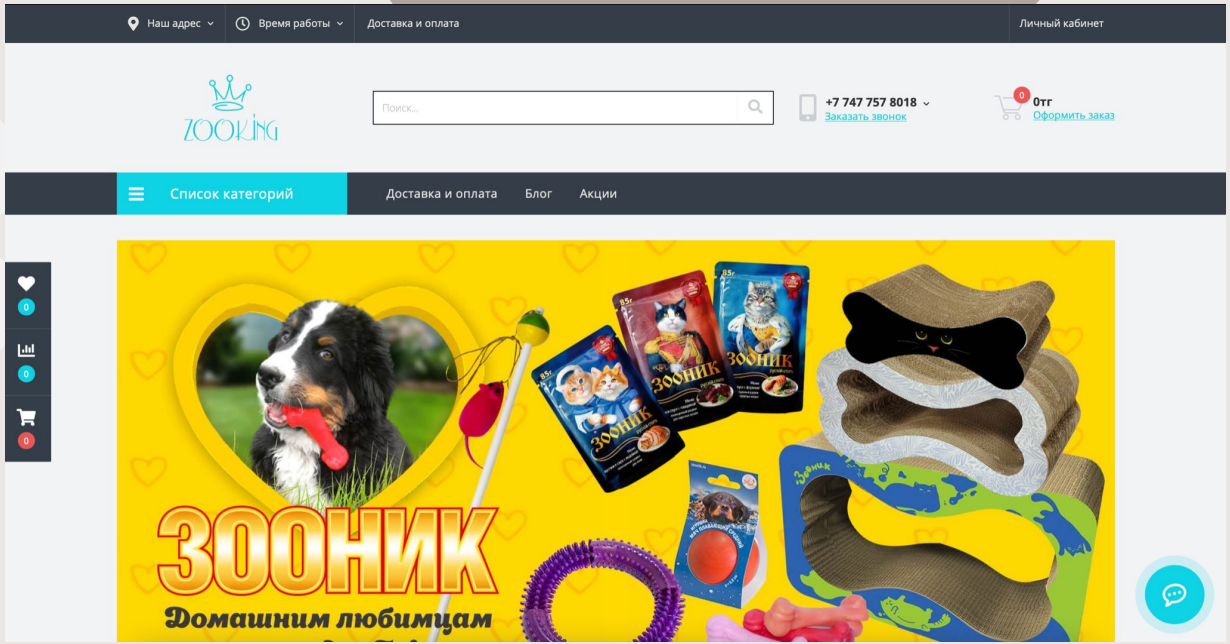
Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser.

Data Sources



zcv.kz

Over than 2400 rows of data were collected from zcv.kz



zoomagazin.kz

Over than 3000+ rows of data were collected from zcv.kz

Methods

Datasets were collected and created using parsers which were written on python 3+ using BeatifulSoup4 library and free data generators

Random Data Generators

Github

For generating huge datasets with 1000+ rows were used free random data generators from github, which were distributed withour limitations *https://github.com/snikitin-de/Generato-datasets*

mockaroo.com

Was used in order to generate data with less than 1000 rows

generatedata.com

Was used as alternative to mockaroo ,due to limitations of the first one, to generate data with less than 1000 rows

Parsers

For both sites were created two universal parsers ,which were written on python with library BeautifulSoup4

```python
import requests as rq
import csv
import uuid
import urllib.request
import shutil
from dotenv import load_dotenv
import os
from os.path import join, dirname
from bs4 import BeautifulSoup

# Dotenv configuration
# All variables are stored in .env
dotenv_path = join(dirname(__file__), '.env')
load_dotenv(dotenv_path)
#Константы
URL = os.environ.get("ZOOMAGAZIN")
HEADERS = {'user-agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_6) Apple
SITE = os.environ.get('SITE') # Ex: 'https://zoomagazin.kz'
FILE = os.environ.get("FILE_NAME") # Ex: 'products.csv'
IMAGES_STORE = os.environ.get("IMAGES_STORE") # Absolute path to the dir where

#CATEGORY = 39 - it just for DB project, you can delete it if necessacy

#Получение html кода страницы
def get_html(url,params=None):
    r = rq.get(url, headers=HEADERS,params=params)
    return r


def download_image(imagePath, imageName):
    response = rq.get(imagePath, stream=True)

    file = open(IMAGES_STORE + imageName, 'wb')

    response.raw.decode_content = True
    shutil.copyfileobj(response.raw, file)
    del response
#Парсинг контента
```

10 000+

rows of data were collected and created

11 tables

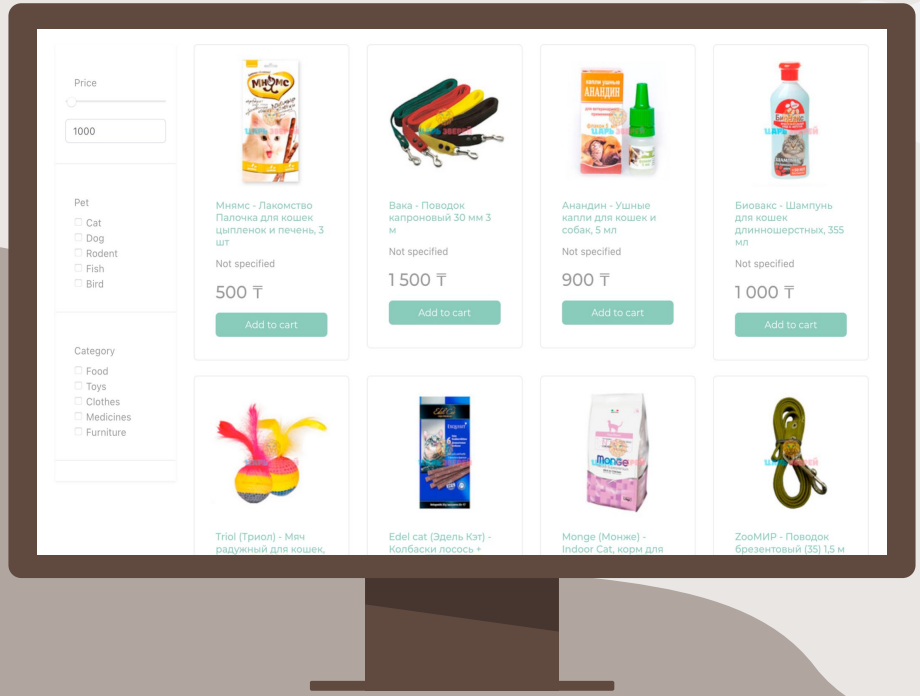
were created in Postgres

24h 37m 23s

is Jupiter’s rotation period

Desktop Software

Fullscreen desktop version



Desktop Software

Fullscreen desktop version

Thank you!