# Автоматизация процессов каталогизации в книжном магазине

## Описание проекта

Создать сервис, который позволит автоматизировать и цифровизировать внутренние процессы каталогизации в книжном магазине с возможностями управления имеющимися книгами и поиска нужных экземпляров.

---

## Принципы разработки

Для разработки соблюдаем принципы SOLID:

- Single Responsibility Principle (Принцип единственной ответственности): Каждый класс в проекте отвечает только за одну задачу.

- Open/Closed Principle (Принцип открытости/закрытости): Систему можно расширять новыми функциями, но при этом минимально модифицировать существующий код.

- Liskov Substitution Principle (Принцип подстановки Лисков): Наследуемые классы должны сохранять поведение базовых классов.

- Interface Segregation Principle (Принцип разделения интерфейсов): Интерфейсы должны быть узкими и специализированными.

- Dependency Inversion Principle (Принцип инверсии зависимостей): Высокоуровневые и низкоуровневые модули зависят от абстракций.

---

## Структура данных

### Книга

- Название

- Жанр

- Автор (авторы)

- Год издания

- ISBN

- Издательство

- Возрастное ограничение

- Количество страниц

- Язык

- Цена

- Количество в наличии

### Автор

- ФИО

- Псевдонимы

- Дата рождения

- Страна

- Жанры

- Список книг автора доступных в магазине

- ID

### Издательство

- Название

- Год основания

- Контактная информация (Телефон, Электронная почта)

- Список изданных книг

- Город

---

## Требования к функционалу

- Добавление новой книги

- Изменение информации о книге

- Поиск книг по ключевому слову

- Фильтрация книг (по автору, жанру и т.д.)

- Выбор множества книг

- Удаление книги или ряда выбранных книг

- Открытие страницы с полной информацией о книге

- Возможность чтения базы из файла

- Возможность сохранения базы в файл

- Открытие страницы автора с его книгами

- Открытие страницы издательства с его книгами

- Авторизация пользователей: консультант и админ

---

## База данных

Используем SQL БД (например, MySQL или MSSQL, обсуждается окончательный выбор). Приложение будет реализовано как web-сайт, связанный с базой данных напрямую. Все изменения будут сразу отображаться в базе данных (синхронное обновление).

---

## Дополнительные вопросы

Обсудить, использовать ли локальный сервер или подключаться к удалённой БД.

---

## Организация работы

### GitHub Workflow

- Репозиторий: основные ветки (main для стабильных релизов, develop для разработки).

- Работа с ветками: Для каждой задачи создаётся отдельная ветка.

- Пул-реквесты проходят ревью перед объединением в develop.

- Документация: вся основная информация по проекту находится в ТЗ и README.

---

## Возможные дополнения проекта

- Учёт движения книг по датам и количеству

- Формирование статистики и отчётности

---

## Тестирование и поддержка

Определить методы тестирования.

---

## Канбан-доска и задачи

- Определить задачи и распределить их по команде.

- Установить приоритеты и сроки для каждой задачи.

- Вести учёт выполненных и текущих задач с помощью канбан-доски (например, GitHub Projects, Trello).