Санкт–Петербургское государственное бюджетное профессиональное образовательное учреждение

«Колледж информационных технологий»

ОТЧЕТ

по производственной практике

**ПМ.01 Разработка модулей программного обеспечения для компьютерных систем**

Специальность 09.02.07 Информационные системы и программирование (программист)

Выполнил

студент гр. 493 \_\_\_\_\_\_\_\_А.Д. Сидоров

Согласовано  
ООО «Омега» \_\_\_\_\_\_\_\_\_С.В. Литвиненко

Руководитель производственной практики \_\_\_\_\_\_\_\_\_\_Н.В. Романовская

Санкт–Петербург   
2021

СОДЕРЖАНИЕ

[Введение 3](#_Toc130242163)

[1. Предметная область 4](#_Toc130242164)

[2. Техническое задание 4](#_Toc130242165)

[3. Формирование алгоритмов разработки программных модулей в соответствии с техническим заданием 4](#_Toc130242166)

[4. Разработка программных модулей в соответствии с техническим заданием 7](#_Toc130242167)

[5. Выполнение отладки программных модулей с использованием специализированных программных средств 16](#_Toc130242168)

[6. Выполнение тестирования программных модулей 16](#_Toc130242169)

[7. Осуществление рефакторинга и оптимизации программного кода 17](#_Toc130242170)

[8. Разработка модулей программного обеспечения для мобильных платформ 17](#_Toc130242171)

[Заключение 18](#_Toc130242172)

# Введение

# Предметная область

На производственной практике в моей организации был выбор предметных областей для прохождения практики, и мной была выбрана предметна область «Магазин котиков», как показано на рисунке 1.1.



Рисунок 1.1 – Выбранная предметная область

Выполненная работа находится по адресу <https://github.com/AntonSidorov1/InterShipOooOmega>.

# Техническое задание

# Формирование алгоритмов разработки программных модулей в соответствии с техническим заданием

В данном разделе описываются алгоритмы, которые я разработал в соответствии с выбранной предметной областью.

Алгоритм авторизации представлен на рисунке 2.1.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 2.1 – Алгоритм авторизации

Алгоритм регистрации представлен на рисунке 2.2.

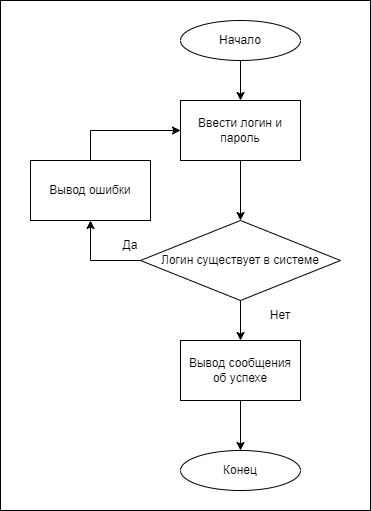


Рисунок 2.2 – Алгоритм регистрации

Алгоритм добавления администратора представлен на рисунке 2.3.

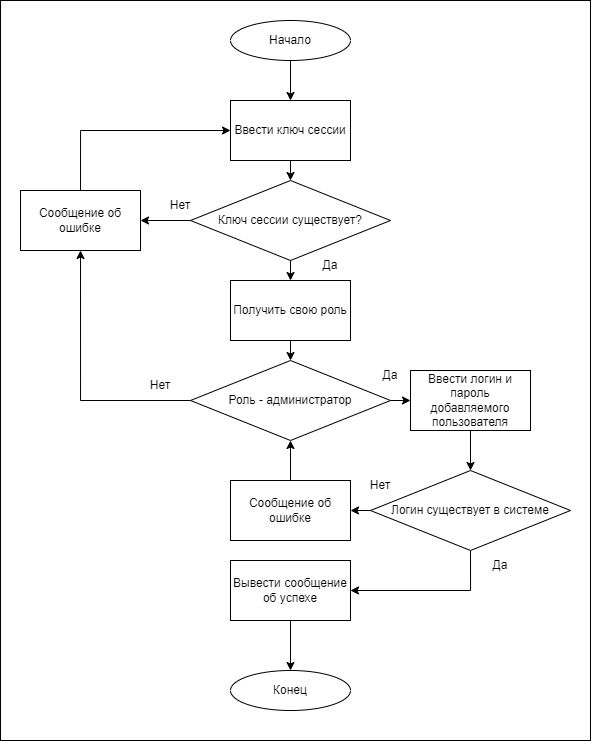


Рисунок 2.3 – Алгоритм добавления администратора

Алгоритм добавления котика представлен на рисунке 2.4.

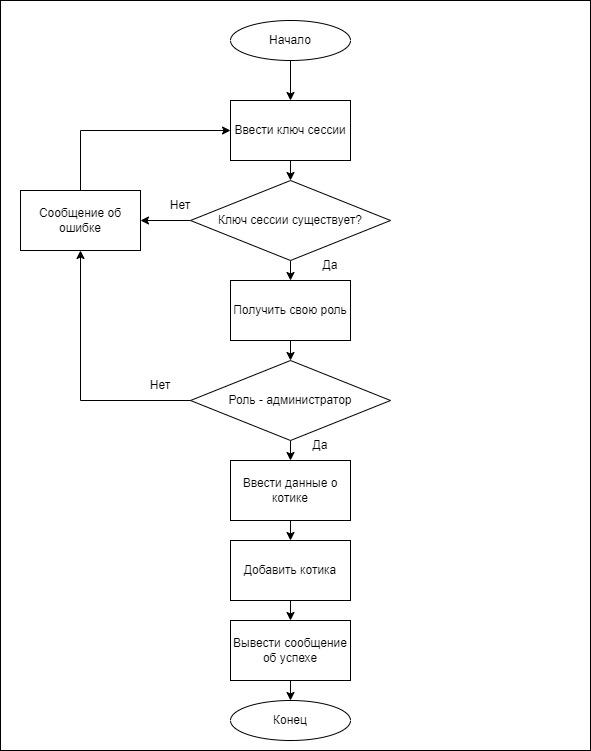


Рисунок 2.4 – Алгоритм добавления котика

Алгоритм удаления котика представлен на рисунке 2.5.

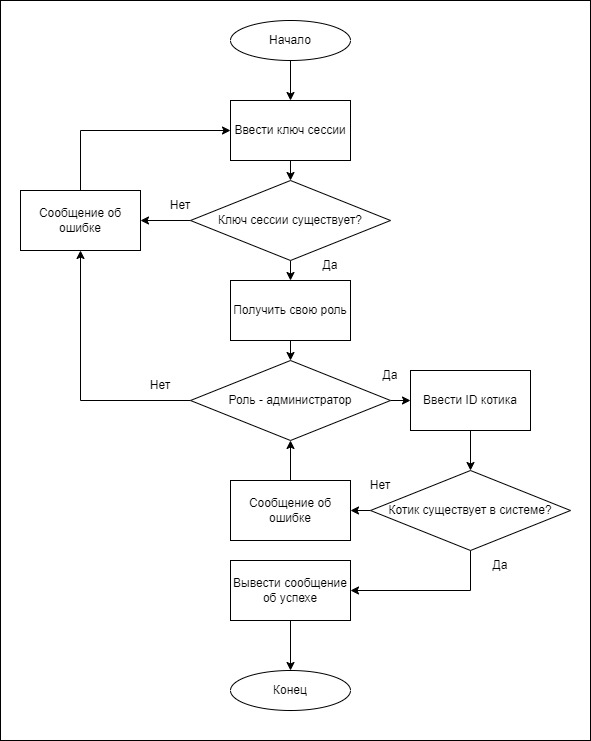


Рисунок 2.5 – Алгоритм удаления котика

# Разработка программных модулей в соответствии с техническим заданием

Данный раздел описывает модули, которые я создал, среди которых присутствует база данных, API.

* 1. Проектирование базы данных

Диаграмма базы данных представлена на рисунке 3.1

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 3.1 – Диаграмма базы данных

В данной диаграмме присутствуют таблицы, описание которых представлено в таблице 3.1.

Таблица 3.1 – Таблицы базы данных.

|  |  |  |  |
| --- | --- | --- | --- |
| **Таблица и её назначение** | **Столбец и его назначение** | **Тип данных в столбце** | **Ограничение в столбце** |
| Role (Роли пользователей в системе) | RoleID (ID роли) | int | Primary key |
| RoleName (Название роли) | Nvarchar(100) | Not Null |
| User (Пользователи в системе) | UserID (ID пользователя) | int | Primary key |
| RoleID (ID роли у пользователя) | int | Not Null,  Foreign key (Role.RoleID) |
| UserLogin (Логин пользователя) | Nvarchar(100) | Not Null, Unique Key |
| UserPassword (Пароль пользователя) | Nvarchar(100) | Not Null |
| Session (сессия пользователя, вошедшего в систему) | SessionKey (ключ сессии) | Char(20) | Primary Key |
| UserID (ID пользователя) | int | Not Null,  Foreign key (User.UserID) |
| CatColor (Цвет котика) | CatColorID (ID Цвета) | Int | Primary Key |
| CatColorName (Название цвета) | Nvarchar(100) | Not Null |
| CatGender (Пол котика) | CatGenderID (ID пола) | Int | Primary Key |
| CatGenderName (Название пола) | char(1) | Not Null |
| CatSpecies (Порода котика) | CatSpeciesID (ID породы) | Int | Primary Key |
| CatSpeciesName (Название породы) | Nvarchar(100) | Not Null |
| CatModel (Модель котика) | CatModelID (ID модели) | Int | Primary Key |
| CatSpeciesID (ID породы) | Int | Not Null,  Foreign Key (CatSpecies. CatSpeciesID) |
| CatColorID (ID цвета) | Int | Not Null,  Foreign Key (CatColor. CatColorID) |
| CatGenderID (ID пола) | Int | Not Null,  Foreign Key (CatGender. CatGenderID) |
| Cat (котик) | CatID (ID котика) | int | Primary key |
| CatModelID (ID модели) | int | Not Null,  Foreign Key (CatModel.CatModelID) |
| CatAge (возраст котика) | Decimal(10, 2) | Not Null |
| Pozition (Позиция котика) | PozitionID (ID позиции) | Int | Primary Key |
| PozitionCatID | int | Not Null,  Foreign Key (Cat.CatID) |
| PozitionCost (стоимость котика) | Decimal(10, 2) | Not Null |
| PozitionDateAdded (Дата добавления котика) | Date | Not Null, Default (Now()) |
| PozitionDateOfChanged (Дата изменения котика) | Date | Not Null, Default (Now()) |

* 1. Разработка базы данных

База данных была разработана на PostgreSQL 13.3. Диаграмма базы данных представлена на рисунке 3.2.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 3.2 – Диаграмма созданной базы данных

* 1. Разработка API

В данном подразделе описаны созданные мной API-функции. API разработано было в приложениях Visual Studio 2022, Visual Studio 2019 и Rider. Тип проекта – .NET ASP.NET Core Web Application / Web API. Язык программирования – C#. Версия dotnet – 7.0.

* + 1. API для строки подключения к базе данных

Данные функции позволяют редактировать строку подключения к базе данных. Этих функций всего 2, которые представлены на рисунке 3.3.

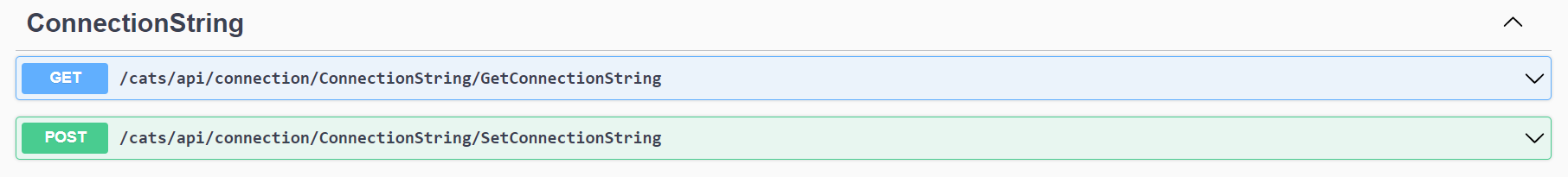


Рисунок 3.3 – Функции API для строки подключения к базе данных

* + 1. API для пользователей

Здесь представлены функции API для работы с пользователями в системе, ролями пользователей и сессиями пользователей.

Функции для работы с ролями

Данные функции предназначены для получения информации о ролях. Эти функции представлены на рисунке 3.4.

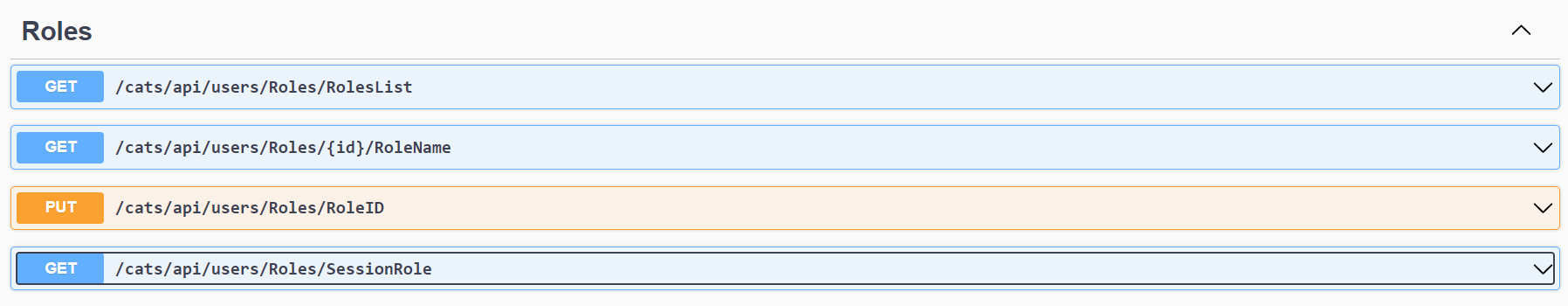


Рисунок 3.4 – Функции API для работы с ролями

Функции для работы с аккаунтами пользователей

Данные функции предназначены для работы с аккаунтами пользователей в системе. Эти функции представлены на рисунке 3.5.

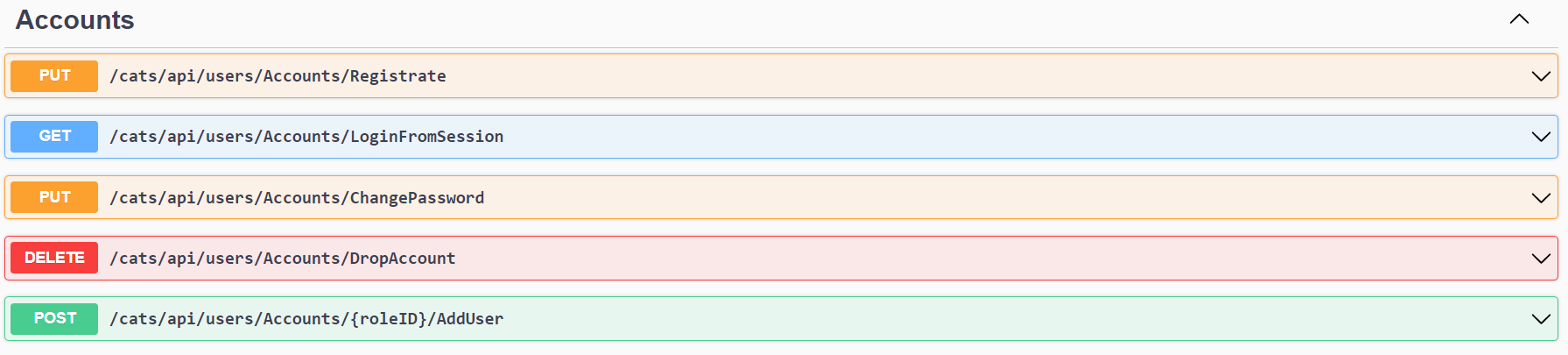


Рисунок 3.5 – API для работы с аккаунтами пользователей

Функции для работы с сессиями пользователей

Данные функции предназначены для работы с сессиями пользователей в системе. Эти функции представлены на рисунке 3.6.

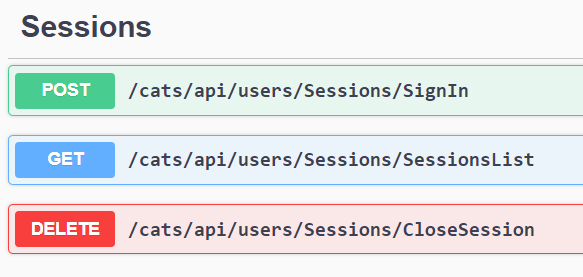


Рисунок 3.6 – Функции API для работы с сессиями пользователей

* + 1. API для котиков

Здесь представлены функции API для работы с котиками и данными о котиках. Каждый котик имеет данные о своей моделе и о своём возрасте. Модель котика имеет данные о поле, возрасте и цвете котика.

Функции для работы с цветами котиков

Данные функции предназначены для работы с цветами котиков. Список этих функций показан на рисунке 3.7.

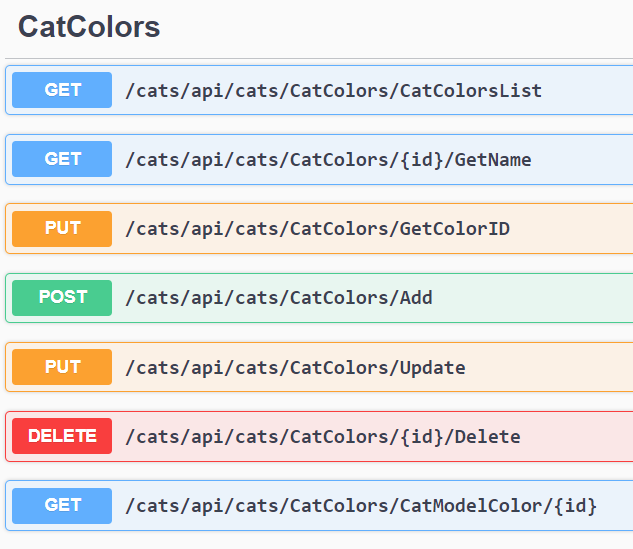


Рисунок 3.7 – Функции API для работы с цветами котиков

Функции для работы с породами котиков

Данные функции предназначены для работы с пародами котиков. Список этих функций показан на рисунке 3.8.

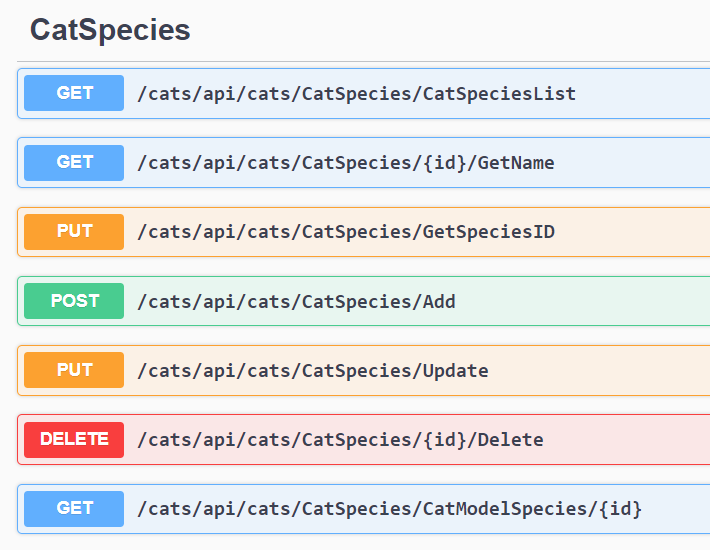


Рисунок 3.8 – Функции API для работы с пародами котиков

Функции для работы с полами котиков

Данные функции предназначены для работы с полами котиков. Список этих функций показан на рисунке 1.9.

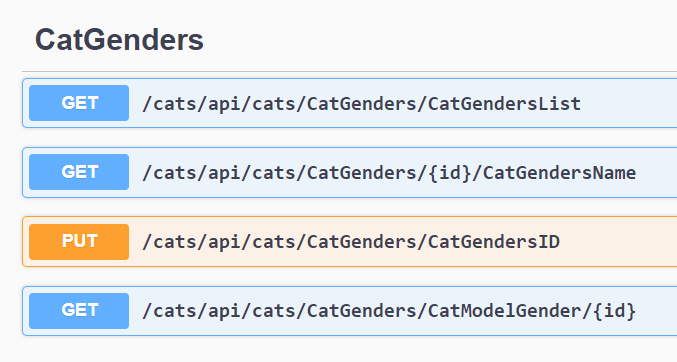


Рисунок 1.9 – Функции API для работы с полами котиков

Функции для работы с моделями котиков

Данные функции предназначены для работы с моделями котиков. Список этих функций показан на рисунке 1.10.

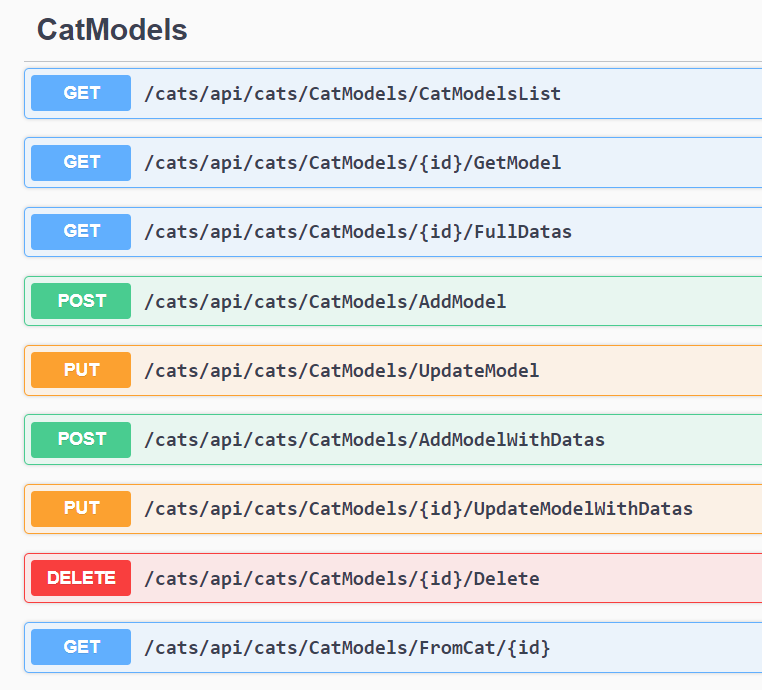


Рисунок 1.10 – Функции API для работы с моделями котиков

Функции для работы с самими котиками

Данные функции предназначены для работы с самими котиками. Список этих функций показан на рисунке 1.11.

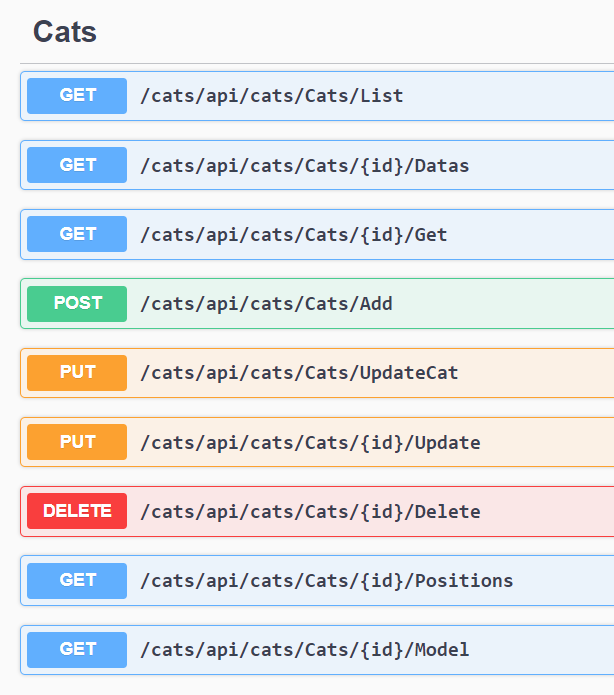


Рисунок 1.11 – Функции API для работы с самими котиками

* + 1. API для работы с позициями котиков

Данные функции предназначены для работы с позициями котиков. Список этих функций показан на рисунке 1.12.

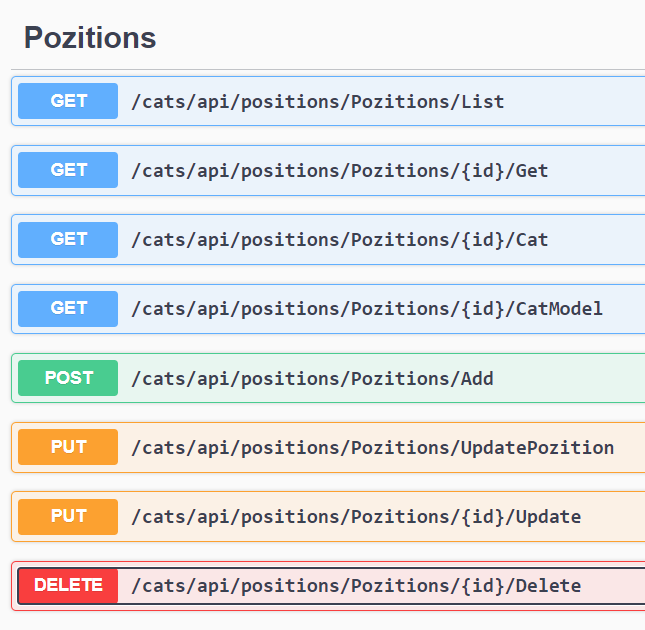


Рисунок 1.12 – Функции API для работы с самими котиками

# Выполнение отладки программных модулей с использованием специализированных программных средств

# Выполнение тестирования программных модулей

# Осуществление рефакторинга и оптимизации программного кода

# Разработка модулей программного обеспечения для мобильных платформ

# Заключение