

## **Рыболов**

Платформа для рыбаков, объединяющая данные о рыбалке, снастях, сообществе и рецептах. Позволяет планировать рыбалку, выбирать снасти, делиться уловами и находить рецепты.

Основные функции:

1. «Когда ловить рыбу» – прогноз клёва, лучшие водоёмы и время лова.
2. База снастей – подбор и проверка совместимости снаряжения.
3. Сообщество – публикация уловов, комментарии, рейтинги.
4. Рецепты – блюда из рыбы с фильтрами по типу улова.
5. «Галерея рыб» – список рыб с их детальным описанием и особенностями ловли.
6. «Галерея водоёмов» – список водоёмов с описанием их местоположения.

## **Описание сервисов**

### **Фронтенд**

JavaScript Angular.

Маршрутизация страницы водоёмов, галереи рыб, снастей, сообщества, рецептов.

Визуализация данных: карты водоёмов, графики клёва, галерея уловов.

Валидация форм (регистрация, добавление улова, комментарии).

Интеграция с картами (Yandex Maps/Google Maps) для отображения водоёмов.

Взаимодействует со всеми сервисами через бэкенд

### **Бэкенд**

Spring Boot, Spring Security.

Обработка REST-запросов, аутентификация через OAuth2.

Логика личного кабинета: сохранение избранных водоёмов, снастей, сборок снаряжения. Здесь же логика «Галерея водоёмов», где описано их расположение и описание, пользователи могут добавлять свои водоёмы после того, как их проверит модерация.

Взаимодействие с БД, и внешними API (погода, карты).

## **Парсер погоды**

Spring Boot.

По открытым API будет братья информация из «Gismeteo» и «Яндекс погода».

Взаимодействует с сервисом «Когда ловить рыбу», «Галерея рыб».

## **Галерея рыб**

Spring Boot

Сервис для представления разных видов рыб с их подробным описанием, методами и особенностями ловли, и фотографиями. Пользователи могут добавлять свои фотографии и добавлять особенности ловли, их проверяет модерация сервиса.

Взаимодействует с бэкендом и БД.

## **Регистрация**

Spring Boot, Spring Security.

Безопасная регистрации. Пользователь вводит адрес электронной почты либо номер телефона, имя, фамилию и пароль с подтверждением. Пароль проверяется на то, что он соответствует подтверждающему паролю, адрес электронной почты проверяется на то, что по нему больше никто не зарегистрирован.

Сервис взаимодействует с бэкендом и БД.

## **Список снастей**

Spring Boot.

Показывает пользователю доступные для покупки снасти, например, удочки, лески, прикормки, блесны, воблеры, экипировку для рыбалки (сапоги, плащи и др.). Множество фильтров, по типу, размеру, количеству, так же есть подбор по рыбе, выбирается вид рыбы и выводятся снасти, которые подходят для её ловли, начиная от удочек, заканчивая приманкой. Есть сравнение товаров.

Сервис взаимодействует с бэкендом, БД, «Галерея рыб».

## **Когда ловить рыбу**

Spring Boot.

Пользователь выбирает водоём и вид рыбы, они проверяются на совместимость, ему выводится дата и время, когда лучше отправиться на рыбалку.

Сервис взаимодействует с бэкендом, БД, парсером погоды, «Галерея рыб».

### **Сообщество**

Spring Boot.

Сервис для опубликования своих уловов. Зарегистрированный пользователь выкладывает пост с удачным уловом, вид рыбы, вес рыбы, её длину, фотографию, на какую приманку и в каком водоёме ловил. Остальные пользователи могут просматривать чужие уловы и оставлять комментарии с советами или вопросами к автору.

Сервис взаимодействует с БД, «Галерея рыб», «Список снастей», «Галерея водоёмов».

### **Рецепты**

Spring Boot.

Сервис для выкладывания рецептов из рыбы. Пользователь вводит тип рыбы, имя рецепта, необходимые ингредиенты, описание приготовления, время приготовления и сложность рецепта, есть возможность добавления фото готового блюда.

Сервис взаимодействует с БД, «Галерея рыб».

### **БД**

PostgreSQL для хранения данных о пользователях, галереи рыб, водоёмов, информацию об уловах, комментариях к ним и рецептам.

### **Мониторинг**

Prometheus, Grafana для визуализации.

Будет отслеживаться время работы сервисов. Время ответа, количество запросов, так же вестись учёт статистики ошибок 4xx и 5xx. Использование процессора, оперативной памяти, базы данных. Также отслеживаться активность пользователей (публикаций/день).

Будет взаимодействовать со всеми сервисами и сохранять данные в собственном формате.

