

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Уральский федеральный университет имени
первого Президента России Б. Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ
Школа бакалавриата

ОТЧЕТ

По проекту
«Разработка интерактивной игры по сохранению диких животных с
использованием ИИ»
по дисциплине «Проектный практикум»

Заказчик: Червяков Дмитрий Андреевич

Куратор: Червяков Дмитрий Андреевич

Студенты команды: хорошая

Клестова Мария Александровна РИ-320942

Ермакова Анна Олеговна РИ-320942

Сурин Владислав Алексеевич РИ-320936

Комаров Максим Сергеевич РИ-320946

Екатеринбург

2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Требования к продукту	5
2 Аналитика	11
3 Архитектура.....	16
4 Процесс разработки.....	18
5 Планирование работ	22
6 Работа тимлида	24
7 Работа аналитика	27
8 Работа frontend-разработчика	30
9 Работа backend-разработчика	35
ЗАКЛЮЧЕНИЕ	38
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	40
ПРИЛОЖЕНИЕ А Основные страницы сайта	41

ВВЕДЕНИЕ

Современное состояние дикой природы требует более широкого внимания общества и научного сообщества к вопросам биоразнообразия. Традиционный мониторинг животных через фотоловушки сопровождается трудоемкой ручной обработкой изображений, что ограничивает эффективность исследований. Использование технологий искусственного интеллекта способно ускорить и автоматизировать этот процесс, однако подобные решения редко адаптируются под практические и образовательные нужды.

Проект решает сразу несколько задач: он повышает экологическую осведомленность пользователей, демонстрирует потенциал ИИ в распознавании животных и облегчает участие широкой аудитории в научно-исследовательской деятельности. Его игровая форма делает процесс вовлекающим и доступным, а также способствует формированию эмпатии к дикой природе.

Целью проекта является разработка адаптивного веб-приложения в формате интерактивной игры, направленной на популяризацию знаний о биоразнообразии и демонстрацию возможностей автоматизации распознавания животных с использованием искусственного интеллекта. Игроки выступают в роли ученых-биологов, анализируя фотографии с фотоловушек для поиска и идентификации животных.

Основные задачи проекта:

- реализация игрового процесса, состоящего из трех этапов: самостоятельного поиска животных, сверки с результатами ИИ и финальной аналитики;
- разработка адаптивного пользовательского интерфейса на основе макетов из Figma [1];
- создание системы подсчета и отображения результатов;
- интеграция базы данных с фотографиями и метаданными;

- обеспечение возможности многопользовательского взаимодействия с системой через современные веб-браузеры.

Продукт предназначен для следующих целевых групп:

- специалисты, работающие с данными с фотоловушек: экологи, биологи, зоологи;
- студенты и преподаватели в области естественных наук, для которых приложение может стать учебным инструментом;
- любители природы, заинтересованные в наблюдении за дикой природой и ее охране.

Веб-приложение может использоваться как в научных и образовательных учреждениях, так и в рамках массовых онлайн-инициатив и экопросветительских проектов.

По завершении проекта планируется получить полнофункциональный веб-продукт, включающий:

- интерактивную игру, доступную через браузеры на всех типах устройств;
- стабильно работающий backend и frontend, реализующие игровой сценарий и обработку данных;
- базу данных с изображениями животных;
- адаптивный интерфейс, соответствующий современным UX/UI [2] стандартам.

Проект продемонстрирует, как игровые механики и ИИ могут быть эффективно объединены для решения прикладных задач в области охраны природы и научного образования.

1 Требования к продукту

Заказчиком проекта является компания «Сбер», и его основное ожидание — это создание веб-приложения в виде интерактивной игры, направленной на популяризацию темы биоразнообразия и демонстрацию возможностей технологий искусственного интеллекта. Ниже приведен перечень требований.

Функциональные требования:

– игра должна состоять из трех этапов:

1) первый раунд: самостоятельный поиск животных на фотографии — пользователь выделяет область с животным и указывает его вид;

2) второй раунд: пользователь видит, как ИИ распознал животное на другом снимке, и оценивает корректность результата с помощью кнопок «Да» / «Нет»;

3) финальный этап: отображение статистики — сравнение пользовательских ответов с данными ИИ, процент совпадений, общее количество найденных животных.

– таймер: каждый игровой раунд ограничен по времени (45 секунд);

– возможность увеличения изображения (zoom) на этапе распознавания;

– кнопка «Пропустить» — пользователь может перейти к следующему заданию;

– система подсчета и отображения результатов:

1) подсчет количества найденных животных;

2) подсчет совпадений с результатами ИИ;

3) отображение итогов.

Технические требования:

– приложение должно работать как веб-сервис, доступный через основные браузеры;

- продукт должен быть кроссбраузерным и поддерживать разрешения от 320px до 2560px (мобильные, планшеты, десктопы);
- интерфейс должен соответствовать предоставленным макетам в Figma;
- дизайн должен быть адаптирован под разные типы устройств без потери функциональности или визуального качества.

Критерии приемки:

- игровой процесс должен соответствовать описанным механикам (3 этапа);
- интерфейс — реализован по макетам из Figma;
- приложение не содержит критических ошибок (зависаний, сбоев при загрузке изображений, некорректных расчетов и т.д.).

Проект ориентирован на три основные группы пользователей:

- специалисты (экологи, биологи, зоологи): ожидают точности, достоверности данных и практической пользы от сравнения с ИИ;
- студенты и преподаватели: заинтересованы в образовательной ценности, простоте интерфейса и доступности;
- любители природы: ценят вовлекающий формат, геймификацию и возможность узнать новое о животных.

Таким образом, программный продукт должен быть:

- интуитивно понятным,
- визуально привлекательным,
- технологически стабильным,
- содержательно наполненным.

План действий для достижения поставленных целей был оформлен в виде бэклога, представленного в таблице 1.

Бэклог представляет собой структурированный список задач, сформулированных на основе требований заказчика и потребностей целевой аудитории. Каждая задача в бэклоге сопровождается приоритетом, статусом,

оценкой трудоемкости, назначением ответственного исполнителя и описанием, что позволило эффективно организовать рабочий процесс и отслеживать прогресс на каждом этапе.

Таблица 1 — Backlog

Контрольная точка	Сprint	Задача	Приоритет	Статус	StoryPoint	Срок	Ответственный	Описание
1	1 (03.03.20 25 — 16.03.20 25)	Создать рабочее пространство	Средний	Завершено	2	03.03.2 025 — 12.03.2 025	Маша	Создать пространство для работы. Графики встреч — опционально.
1	1 (03.03.20 25 — 16.03.20 25)	Аналитика ЦА	Средний	Завершено	8	10.03.2 025 — 15.03.2 025	Аня	Провести анализ ЦА, их потребности и ожидания.
1	1 (03.03.20 25 — 16.03.20 25)	Анализ аналогов	Средний	Завершено	8	10.03.2 025 — 15.03.2 025	Маша	Посмотреть существующие аналоги, +-, сравнение, вывод.
1	1 (03.03.20 25 — 16.03.20 25)	Проблематика	Высокий	Завершено	3	03.03.2 025 — 12.03.2 025	Аня	Сформулировать проблематику бедолагу(
1	1 (03.03.20 25 — 16.03.20 25)	Создать репозитории GitHub	Средний	Завершено	2	05.03.2 025 — 09.03.2 025	Влад, Максим	Создать и настроить структуру репозиториев для фронта и бэкенда.
1	1 (03.03.20 25 — 16.03.20 25)	Проектирование БД	Средний	Завершено	10	05.03.2 025 — 10.03.2 025	Влад	Создать бд.
1	1 (03.03.20 25 — 16.03.20 25)	Создание основной функциональности	Высокий	Завершено	20	03.03.2 025 — 16.03.2 025	Максим	По клику на животное получить варианты ответов.

Продолжение таблицы 1

Контрольная точка	Сprint	Задача	Приоритет	Статус	StoryPoint	Срок	Ответственный	Описание
1	1 (03.03.2025 — 16.03.2025)	Верстка по макету	Средний	Завершено	20	03.03.2025 — 16.03.2025	Максим	На основе дизайна в фигме сделать верстку, стилизовать максимально адаптивно.
1	1 (03.03.2025 — 16.03.2025)	Реализация механики первого этапа игры (поиск животных)	Средний	Завершено	25	03.03.2025 — 16.03.2025	Влад	—
1	2 (17.03.2025 — 30.03.2025)	User Stories	Высокий	Завершено	8	17.03.2025 — 25.03.2025	Аня	Написать юзер сторис 5 штук минимум + трудозатраты .
1	2 (17.03.2025 — 30.03.2025)	Составление ТЗ	Низкий	Завершено	10	17.03.2025 — 25.03.2025	Маша	Сформировать итоговое техническое задание.
1	2 (17.03.2025 — 30.03.2025)	Реализация второго этапа игры (сравнение с ИИ)	Средний	Завершено	20	17.03.2025 — 30.03.2025	Влад	—
1	2 (17.03.2025 — 30.03.2025)	Доработка функциональности	Низкий	Завершено	16	17.03.2025 — 30.03.2025	Максим	Доработка основной функциональности!!
1	3 (31.03.2025 — 13.04.2025)	Разработка адаптивного дизайна для телефона	Средний	Завершено	6	31.03.2025 — 05.04.2025	Влад	Создание мобильных макетов в фигме, упрощение элементов наверное.

Продолжение таблицы 1

Контрольная точка	Сprint	Задача	Приоритет	Статус	StoryPoint	Срок	Ответственный	Описание
1	3 (31.03.2025 — 13.04.2025)	Разработка адаптивного дизайна для телефона	Средний	Завершено	6	31.03.2025 — 05.04.2025	Влад	Создание мобильных макетов в фигме, упрощение элементов наверное.
1	3 (31.03.2025 — 13.04.2025)	Верстка адаптивного дизайна	Средний	Завершено	15	05.04.2025 — 13.04.2025	Максим	Реализация адаптивности на фронтенде.
2	4 (14.04.2025 — 27.04.2025)	Оптимизация запросов к БД	Низкий	Завершено	10	14.04.2025 — 18.04.2025	Влад	Повышение скорости работы бд.
2	4 (14.04.2025 — 27.04.2025)	Связка фронта и бэка	Средний	Завершено	20	18.04.2025 — 27.04.2025	Влад, Максим	Настройка взаимодействия, тестирование запросов и корректности данных.
2	5 (28.04.2025 — 11.05.2025)	Развертывание бэка на сервере	Высокий	Завершено	8	28.04.2025 — 11.05.2025	Влад	Настройка хостинга, бд, деплой...
2	5 (28.04.2025 — 11.05.2025)	Развертывание итогового фронта	Средний	Завершено	6	28.04.2025 — 11.05.2025	Максим	Деплой на Vercel)
3	6 (12.05.2025 — 25.05.2025)	Проведение нагрузочного тестирования	Средний	Завершено	7	12.05.2025 — 15.05.2025	Влад	Анализ производительности системы при высоких нагрузках.

Продолжение таблицы 1

Контролльная точка	Сprint	Задача	Приоритет	Статус	StoryPoint	Срок	Ответственный	Описание
3	6 (12.05.2 025 — 25.05.2 025)	Проведение нагрузочного тестирования	Средний	Завершено	7	12.05.2 025 — 15.05.2 025	Влад	Анализ производительности системы при высоких нагрузках.
3	6 (12.05.2 025 — 25.05.2 025)	Проведение юзабилити-тестирования	Низкий	Завершено	12	12.05.2 025 — 15.05.2 025	Аня	Тестирование удобства интерфейса и игрового процесса, сбор ос.
3	6 (12.05.2 025 — 25.05.2 025)	Правки по результатам тестирования	Средний	Завершено	20	15.05.2 025 — 25.05.2 025	Влад, Максим	Исправление выявленных багов, доработка интерфейса.
3	7 (26.05.2 025 — 08.06.2 025)	Подготовка финального отчета	Низкий	Завершено	15	01.06.2 025 — 08.06.2 025	Маша	Создание итогового отчета о проделанной работе.
3	7 (26.05.2 025 — 08.06.2 025)	Подготовка документации	Низкий	Завершено	10	26.05.2 025 — 08.06.2 025	Влад, Максим	Написание технической и пользовательской документации.

2 Аналитика

Проблематика проекта заключается в недостаточной автоматизации и популяризации мониторинга диких животных с использованием фотоловушек.

В настоящее время обработка фотоматериалов требует значительных временных и человеческих ресурсов от специалистов, таких как зоологи, экологи и биологи. Существующие решения по распознаванию животных на изображениях, полученных с фотоловушек, пока не нашли широкого применения и зачастую слабо адаптированы для задач полевых исследований.

Ключевая задача проекта заключается в демонстрации того, как современные цифровые технологии и искусственный интеллект могут способствовать наблюдению за дикой природой.

Благодаря игровой форме, пользователи обучаются распознавать животных на фотографиях с фотоловушек. Это не только делает процесс интересным и познавательным для широкой аудитории, но и подчеркивает зоологам и экологам полезность таких решений в их реальной деятельности.

Целевая аудитория проекта была определена заказчиком и включает три ключевые группы пользователей:

1) Ученые: биологи, зоологи, экологи. Это специалисты, которые непосредственно занимаются сбором, анализом и интерпретацией данных о дикой природе, в том числе на основе изображений с фотоловушек. Для этой группы особенно актуальны:

- точность и достоверность информации, представленной в игре;
- возможность наблюдать, как искусственный интеллект обрабатывает данные, сопоставляя свои ответы с пользовательскими;
- использование продукта как демонстрационной платформы при обучении, проведении лекций и общественных мероприятий;

2) любители природы и экологии — это широкая аудитория, включающая как заинтересованных граждан, так и активистов, волонтеров, участников экодвижений. Для этой группы важно:

- наличие увлекательного и визуально привлекательного игрового процесса;
- возможность узнавать новое о животных и природе в непринужденной форме;
- получение позитивного опыта от взаимодействия с реальными данными (фото с фотоловушек);
- развитие экологической эмпатии через интерактивную игру.

3) студенты и преподаватели в области биологии и естественных наук.

В эту категорию входят:

- школьники и студенты вузов, изучающие естественнонаучные дисциплины;
- преподаватели, использующие современные интерактивные и геймифицированные методы обучения.

Потребности этой группы:

- удобный, понятный и логичный интерфейс;
- наличие обучающей составляющей в игровом процессе;
- возможность использования игры в образовательных курсах как дополнение к теоретическим материалам;
- визуализация данных и демонстрация работы ИИ как часть учебной программы.

Таким образом, продукт должен быть одновременно информативным, образовательным и геймифицированным, чтобы удовлетворить потребности обеих частей целевой аудитории.

Для обзора аналогов были рассмотрены существующие продукты и платформы, частично решающие сходные задачи — от распознавания

животных до вовлечения пользователей в проекты по защите природы; они представлены в таблице 2.

Таблица 2 — Анализ аналогов

Аналог	Достоинства	Недостатки	Отличие от нашего продукта
Zooniverse (Snapshot Serengeti)	Большая пользовательская база, богатая коллекция данных.	Нет составляющей, визуальная вовлеченность, интерфейс.	У игровой слабая геймифицированный опыт, больше вовлечения через механику игры.
Wildlife Insights	Сильная ИИ-система, поддержка Google, автоматическая сортировка снимков, полезно для ученых.	Сложный вход, интерфейс ориентирован только на специалистов, не подходит для широкой аудитории.	Мы предлагаем доступную форму через игру и визуальное взаимодействие.
iNaturalist	Огромное комьюнити, элементы геймификации (бейджи, уровни), сильное распознавание ИИ.	Нет реальных фото фотоловушек, пользователи сами делают снимки.	Наш фокус — именно фотоловушки и процесс их анализа как у учёных.
Seek by iNaturalist	Обучающее приложение, распознает животных/растения через камеру, работает в реальном времени.	Не работает с архивными снимками, нет аналитики и сравнений с ИИ.	Мы сравниваем действия человека и ИИ, + архивные данные.
Наш проект	Геймификация, простота, вовлеченность, научная ролевая обучающий сравнение с ИИ, адаптивность под любые устройства.	Пока нет элементов сообщества, нужна работа над реальной точностью ИИ.	Уникальное сочетание обучения, игры и демонстрации ИИ в одном продукте.

Проанализировав существующие решения в области мониторинга дикой природы с использованием фотоловушек, можно выделить несколько ключевых выводов:

1) недостаток интеграции с реальной практикой: Большинство существующих решений ограничены экспериментальными исследованиями

или имеют низкий уровень адаптации для повседневного использования учеными и экологами.;

2) отсутствие геймификации и образовательных элементов: Аналоги фокусируются только на функциональности распознавания, не предлагая пользователям увлекательный и познавательный опыт. В то время как наш проект сочетает в себе обучающий элемент и игровые механики, что делает его более доступным и интересным для широкой аудитории;

3) малое внимание к пользовательскому интерфейсу и доступности: в большинстве аналогов интерфейсы сложно адаптируются для пользователей с разным уровнем подготовки, что снижает их эффективность. Наше приложение акцентирует внимание на простоте интерфейса и доступности для всех категорий пользователей, включая студентов, биологов и просто любителей природы.

Преимущества нашего решения:

- доступность: сайт без регистрации, без установки, работает на любых устройствах;
- обучение через действие: игроки интуитивно понимают, как работает ИИ и каково быть в роли зоолога;
- геймификация: таймер, результаты, очки, сравнение с ИИ делают процесс увлекательным;
- актуальность: реальные данные, реальные животные, реальные проблемы;
- двойная ценность: интересно пользователям, полезно профессионалам;
- масштабируемость: решение может развиваться — добавляться новые животные, регионы, сценарии (например, "ночные животные", "редкие виды" и т.п.).

Таким образом, наше решение имеет явное преимущество в плане интеграции образовательных элементов, улучшенной точности ИИ и адаптивности интерфейса, что позволяет создать более привлекательный и

полезный инструмент для мониторинга дикой природы и обучения пользователей.

3 Архитектура

Проект реализован по принципу трехуровневой архитектуры (клиент – сервер – база данных). Это обеспечивает четкое разделение обязанностей, облегчает масштабирование, модификацию и поддержку продукта.

Основной стек:

- frontend (клиент): React [3] + TypeScript [4];
- backend (сервер): Java [5] + Spring Framework [6];
- база данных: PostgreSQL [7].

Основные компоненты и их взаимодействие:

1) клиентская часть (Frontend):

- запуск и управление игровыми раундами;
- отображение изображений и интерфейса выделения объектов;
- визуализация финальной аналитики;
- поддержка нескольких языков интерфейса.

2) серверная часть (Backend):

- генерация уровней;
- хранение и обработка пользовательских результатов;
- сравнение ответов с эталонными значениями;
- отправка UI-текстов и справочной информации;
- обеспечение API для фронтенда.

Краткая структура базы данных представлена в таблице 3.

Таблица 3 — База данных

Таблица	Назначение
animals	Список всех животных.
animal_name	Названия животных на разных языках.
animal_description	Описание животных на разных языках.
levels_first_round	Данные для первого раунда игры (изображение, координаты животного, правильный ответ).

Продолжение таблицы 3

Таблица	Назначение
levels_second_round	Данные для второго раунда (включает как координаты, так и предполагаемый ответ ИИ).
level_animals	Дополнительные животные, встречающиеся на уровне.
ui_texts и multilingual_ui_texts	Многоязычные строки для интерфейса.

Выбор описанной архитектуры обоснован следующими факторами:

- гибкость и масштабируемость: легко расширять логику или добавлять новые уровни, языки, справки и функции без полной переработки кода;
- надежность и целостность данных: PostgreSQL обеспечивает хорошую работу с внешними ключами и целостностью данных;
- поддержка мультиязычности: реализована через отдельные таблицы для текстов интерфейса и информации о животных.

4 Процесс разработки

В рамках проекта была выбрана гибкая методология разработки, приближенная к Scrum, с итеративным подходом:

- работа велась в спринтах (итерациях), продолжительностью около 1–2 недель;
- в начале каждой итерации проводилось планирование, декомпозиция задач и постановка целей;
- в конце — демонстрация результата заказчику и далее на контрольной точке;
- использовались инструменты визуального управления задачами (бэклог, доска Yougile);
- ответственность распределялась внутри команды по ролям: тимлид, аналитик, frontend и backend разработчики.

Такая организация позволила быстро адаптироваться к изменениям, в том числе к доработкам со стороны заказчика, и поддерживать прозрачность прогресса.

Работа над проектом включала следующие ключевые этапы:

- a) сбор и анализ требований:
 - были изучены материалы, предоставленные заказчиком (техническое задание, дизайн-макеты из Figma, описание целевой аудитории);
 - выделены основные функциональные блоки: игровой процесс (три раунда), работа с изображениями, сравнение с ИИ, мультиязычность;
 - проанализированы потребности трех целевых групп: ученых, любителей природы и преподавателей/студентов, что позволило сбалансировать образовательную и игровую составляющие;
 - сформирован предварительный бэклог задач и определены приоритеты.
- b) разработка архитектуры:
 - определена общая архитектура;

- выбран стек технологий: React + TypeScript (frontend), Java + Spring (backend), PostgreSQL (база данных);
- определены основные сущности базы данных;
- спроектированы основные API-эндпоинты для взаимодействия клиента и сервера;
- продуманы механизмы локализации, хранения изображений и логика работы с координатами животных.

с) верстка и реализация интерфейса:

- интерфейс разрабатывался на основе макетов из Figma, предоставленных заказчиком;
- были реализованы адаптивные стили для отображения на мобильных устройствах, планшетах и десктопах;
- интерфейс, который включает: стартовую страницу с выбором языка; страницу с выбором режима; игровой экран для первого и второго раундов; финальный экран со статистикой;

д) создание backend-логики:

- реализованы REST API для всех операций, включая: получение информации об уровнях и животных; отправку результатов пользователя; генерацию финального отчета;
- разработаны обработчики изображений и работы с координатами;
- добавлена поддержка мультиязычности.

е) наполнение базы данных:

- в базу данных были загружены: список животных с названиями на двух языках; изображения уровней для обоих раундов; координаты животных на изображениях;

ф) интеграция компонентов:

- связь между frontend и backend;
- проведена отладка процессов обмена данными: корректная передача координат, обработка JSON-ответов [8], отображение ошибок;

– обеспечена совместимость всех элементов на разных устройствах и браузерах.

g) тестирование и выявление ошибок:

- тестирование проводилось вручную на каждом этапе;
- были выявлены и устраниены ошибки в координатах, отображении текста;
- протестирована мультиязычность, адаптивность, стабильность игры при переключении режимов.

h) подготовка к демонстрации и защите

- созданы презентационные материалы: слайды, устный доклад;
- финализированы отчеты по задачам, оформлен итоговый отчет по проекту;
- проведена проверка готовности продукта: отсутствие критических ошибок, соответствие требованиям заказчика, демонстрационная стабильность.

Тестирование проводилось поэтапно, по мере готовности отдельных компонентов. Это представлено в таблице 4.

Таблица 4 – Этапы тестирования

Этап	Что тестировалось	Результаты
Разметка интерфейса	Корректность отображения, адаптивность	Выявлены мелкие расхождения с макетами Figma, устраниены
Первый раунд	Работа с изображениями, корректность выбора животного	Были случаи некорректного отображения координат названий животных, исправлено
Второй раунд	Отображение данных ИИ, кнопки подтверждения	Были случаи некорректного отображения координат названий животных, исправлено
Мобильная версия	Адаптивность интерфейса	Переработано расположение элементов на странице

Благодаря итеративному подходу и раннему тестированию команда смогла своевременно выявить и устранить ключевые недочеты, не дожидаясь финальной сборки. Это позволило к моменту завершения проекта иметь стабильный и рабочий продукт, отвечающий требованиям как заказчика, так и конечных пользователей. То, как выглядит продукт сейчас, можно посмотреть в приложении А.

5 Планирование работ

Разработка проекта велась командой из четырех участников с четким распределением ролей и обязанностей. В команде были следующие специалисты:

- Маша — тимлид;
- Аня — аналитик;
- Максим — frontend-разработчик;
- Влад — backend-разработчик.

Процесс разработки строился по итерационной модели, приближенной к Scrum, и включал:

- постановку целей на каждый спринт;
- декомпозицию задач;
- назначение ответственных за каждую задачу;
- еженедельные созвоны;
- использование онлайн-доски для отслеживания задач и статуса их выполнения;
- ведение таск трекера с задачами, распределенными по дедлайнам и приоритетам.

Для планирования и координации использовались следующие инструменты:

- Yougile — визуальная доска задач;
- Git / GitHub — контроль версий и совместная работа с кодом;
- Telegram — ежедневная коммуникация внутри команды;
- Figma — доступ к макетам интерфейса;
- Teams — еженедельные созвоны.

Распределение задач между участниками можно увидеть в таблице 5.

Таблица 5 — Распределение обязанностей в команде

Участник	Роль и основные обязанности
Маша (тимлид)	Планирование этапов, декомпозиция задач, контроль сроков, координация взаимодействия, общение с заказчиками, работа с базой данных, аналитика, презентация к контрольным точкам, финальная презентация, отчет.
Аня (аналитик)	Работа с техническим заданием, описание требований, аналитика, работа с базой данных.
Максим (frontend)	Верстка интерфейса по макетам Figma, реализация логики игры на React, адаптивность для разных устройств, интеграция с API.
Влад (backend)	Реализация серверной логики на Java Spring, работа с базой данных (PostgreSQL), генерация уровней.

Преимущества такого подхода:

- позволил четко разделить зоны ответственности;
- обеспечил параллельную работу над разными частями проекта;
- способствовал оперативной коммуникации;
- обеспечил прозрачность прогресса и контроль над сроками выполнения задач.

6 Работа тимлида

На протяжении всего семестра Маша исполняла обязанности тимлида команды по разработке веб-приложения в формате игры, посвященной биоразнообразию. Ее деятельность охватывала как стратегическое планирование, так и практическое участие в реализации ключевых компонентов проекта.

С первых дней работы над проектом Маша взяла на себя роль координатора и организатора команды. Одной из ее ключевых задач было формирование общего плана разработки с учетом требований заказчика и конечных сроков. Она распределила проект по этапам, обозначив приоритеты, зависимости и точки контроля.

Большое внимание уделялось декомпозиции задач — Маша обеспечивала логичное и последовательное разбиение крупных блоков работ (например, игровой логики, отображения интерфейса, настройки базы данных) на отдельные задачи, которые можно эффективно распределить между участниками команды.

В ходе выполнения проекта Маша регулярно проводила контрольные проверки статуса выполнения задач, обновляла трекер прогресса и вела документацию по проектным активностям. Она следила за соблюдением сроков, оперативно реагировала на возможные риски и задержки. В случае необходимости Маша перераспределяла ресурсы команды и адаптировала план работы, чтобы сохранить общий темп и не выйти за рамки дедлайнов.

Будучи связующим звеном между участниками команды, Маша играла ключевую роль в обеспечении прозрачной и эффективной коммуникации. Она следила за тем, чтобы каждый участник понимал свои задачи и имел доступ ко всей необходимой информации. При возникновении технических или организационных вопросов Маша выступала медиатором, помогая найти наилучшее решение.

Маша также выполняла функцию представителя команды во взаимодействии с заказчиком. Она участвовала в обсуждениях и презентациях на «контрольных точках» Сбера, уточняла требования и при необходимости вносила изменения в проектный план. Ее задача заключалась в том, чтобы обеспечить максимальное соответствие продукта ожиданиям заказчика, сохраняя при этом реалистичность и достижимость задач в рамках учебного процесса.

В рамках аналитической работы Маша провела сравнительный анализ существующих решений и аналогов в смежной области — таких как Zooniverse, iNaturalist, Seek и других. Она выявила ключевые недостатки у конкурирующих решений: отсутствие геймификации, сложность интерфейса, слабая вовлеченность пользователей, а также низкий уровень обратной связи. На основании этого анализа были сформулированы преимущества и уникальные особенности нашего проекта.

Этот анализ лег в основу дальнейшего обоснования архитектурных и интерфейсных решений, а также был использован при защите концепции проекта перед заказчиком и в отчетной документации.

Помимо организационно-управленческих задач, Маша принимала участие и в технической части проекта, в частности, в наполнении базы данных.

Одной из ключевых зон ответственности Маши была подготовка презентационных материалов к контрольным точкам проекта. Она занималась:

- составлением структуры и сценария доклада;
- оформлением презентаций;
- подготовкой команды к выступлению и демонстрации продукта.

Финальный этап семестра включал написание итогового проектного отчета, которым также занималась Маша.

Роль Маши как тимлида оказалась критически важной для успешной реализации проекта. Ее деятельность охватывала все ключевые аспекты

управления командой и проектом — от стратегического планирования до практического участия в аналитике и написании отчетной документации. Благодаря ее вовлеченности, ответственности и системному подходу команда смогла своевременно и качественно реализовать поставленные цели, представить функциональный и законченный цифровой продукт, соответствующий запросам заказчика.

7 Работа аналитика

Аня выполняла роль аналитика в проектной команде. В течение всего семестра она последовательно занималась определением концепции проекта, формализацией требований, работой с игровыми данными и наполнением системы. Ее работа обеспечила логическую и содержательную основу для реализации всех ключевых компонентов веб-приложения.

На первом этапе проекта Аня занималась системной аналитикой и подготовкой всех необходимых документов и решений, на которых в дальнейшем строилась разработка.

Ключевые задачи:

- определение проблематики и цели проекта: Аня сформулировала основную проблему — недостаточную автоматизацию и популяризацию мониторинга диких животных с использованием фотоловушек. Была чётко обозначена цель — создание веб-игры, которая демонстрирует потенциал ИИ в распознавании животных и одновременно вовлекает пользователей;
- анализ целевой аудитории (ЦА): на основе данных от заказчика Аня детализировала ЦА, выделив ключевые потребности и ожидания: развлекательный и образовательный характер игры, доступность интерфейса и значимость контента;
- написание пользовательских историй (user stories): разработаны сценарии использования, описывающие поведение типичных пользователей (эколог, студент, школьник, активист), что позволило разработчикам понять структуру взаимодействия с продуктом;
- формализация требований: Аня подготовила структурированный документ с функциональными и нефункциональными требованиями: от требований к адаптивному дизайну и поддержке мультиязычности до специфики обработки координат животных.

Этот этап заложил прочный фундамент для дальнейших стадий разработки и позволил команде двигаться с четким пониманием целей и критериев успеха.

На втором этапе проектирования, когда началась активная работа с игровым контентом, Аня перешла от стратегической аналитики к оперативной аналитике и подготовке данных для наполнения уровней.

Основные задачи:

– обработка фотографий с фотоловушек: Аня получила от заказчика подборку изображений, на которых необходимо было определить местоположение животных. Она выполняла ручной визуальный анализ фотографий и выделяла координаты объектов (животных), используя вспомогательный инструмент, созданный фронтенд-разработчиком специально для этих целей;

– фиксация координат животных: по каждому изображению фиксировались: координаты левого верхнего угла области с животным (X и Y), ширина и высота выделенной области. Эти данные необходимы для реализации логики первого игрового раунда, где пользователь сам должен обнаружить животное на изображении;

– создание файлов JSON: для оптимизации работы и последующего наполнения базы данных, Аня сформировала отдельные JSON-файлы, содержащие: ссылки на изображения; координаты; правильные ответы; список доступных вариантов выбора. Эти файлы служили промежуточным форматом между визуальной разметкой и интеграцией в backend, обеспечивая чистую, валидную структуру данных для базы. Объем проделанной работы: в общей сложности было обработано более 50 изображений, для каждого из которых были определены точные координаты и создан соответствующий JSON-файл. Это обеспечило основу для генерации раундов игры и взаимодействия пользователя с системой.

Этот этап потребовал внимательности, аналитического мышления и координации с фронтенд-разработчиком — без корректных координат невозможна реализация механики взаимодействия в игре.

На заключительном этапе проекта Аня занималась непосредственным наполнением базы данных, участвуя в интеграции контента в продукт.

Ключевые задачи:

- работа с интерфейсом Swagger [9]: по инструкции, предоставленной бэкенд-разработчиком, Аня заполняла базу данных через Swagger UI — интерактивную документацию, позволяющую добавлять и тестировать данные через HTTP-запросы;
- добавление данных о животных: Аня внесла в базу порядка 30 видов животных на двух языках — русском и английском. Эти данные стали основой для справочной информации, отображаемой в игре, а также частью логики сравнения ответов;
- формирование JSON-файлов для игровых раундов: используя ранее собранные координаты и презентации с вариантами, Аня совместно с тимлидом Машей: создала JSON-файлы для каждого уровня; распределила объем работы (46 уровней для каждого из двух раундов — всего 92 файла); вручную проверила структуру и логику каждого файла перед загрузкой в систему;
- проверка отображения и результатов: после загрузки Аня верифицировала отображение уровней, убедившись, что данные корректно загружаются и доступны пользователю.

Аня проделала всестороннюю и критически важную работу по содержательной части проекта. Ее обязанности охватывали:

- стратегическое проектирование и анализ;
- работу с визуальным контентом и координатами;
- формирование и структурирование JSON-данных;
- наполнение базы данных и обеспечение корректности отображения.

Благодаря Ане в игре реализован точный механизм взаимодействия с изображениями, подготовлена обучающая часть (справки о животных) и обеспечена связность всех уровней. Её работа напрямую связана с игровым опытом пользователя и точностью сравнения с ИИ.

8 Работа frontend-разработчика

Максим занимал ключевую техническую позицию в проекте — он отвечал за разработку клиентской части веб-приложения, реализующего весь пользовательский игровой процесс. Его зона ответственности включала: архитектуру интерфейса, визуальное взаимодействие, обработку изображений, адаптивность, интеграцию с сервером и доработку вспомогательных инструментов.

На протяжении всего семестра Максим активно участвовал в разработке и непрерывно развивал функциональность системы, внося улучшения по мере эволюции требований. Основные достижения на первом этапе:

- создание каркаса React-приложения: Максим разработал структуру компонентов, необходимых для реализации пошаговой логики игры: экранов раундов, стартовых и финальных страниц, логики перехода между ними. Это стало основой всего фронтенда;
- реализация игрового сценария: в первом раунде пользователь кликает на изображение и указывает животное. Максим реализовал алгоритм, проверяющий, попадает ли клик пользователя в нужную область (по координатам, полученным с сервера). Это ядро первого раунда;
- работа с изображениями и координатами: вместо заранее размеченных изображений, Максим реализовал динамическое отображение выделенной области на картинке, что позволило: исключить необходимость хранить десятки версий одного изображения; упростить масштабирование и наполнение уровней;
- функциональность двух игровых режимов: были реализованы оба раунда игры: самостоятельный поиск и проверка с ИИ. Также появилась логика «пропуска» уровня, таймер и визуальная обратная связь (подсветка кнопок правильного/неправильного ответа);

- масштабирование изображений: для улучшения UX была реализована возможность открывать изображения в полноэкранном режиме с сохранением логики выделения животных;
- развертывание демо-версии: для демонстрации хода работ приложение было развернуто на платформе Vercel, что обеспечило постоянный доступ к актуальной версии проекта.

На втором этапе Максим выполнил эти задачи:

- интеграция с backend API: было удалено использование временных моков и подключено реальное API: загрузка уровней; отправка пользовательских ответов; получение справочной информации и текстов интерфейса;
- докеризация: Максим создал: Dockerfile [10] для контейнеризации фронтенда; docker-compose.yml для сборки всей среды с БД; систему инициализации с предварительно загруженным дампом базы данных. Это обеспечило повторяемость развертывания, удобную отладку и универсальность для любых окружений;
- доработка UI и логики UX: внедрены промежуточные экраны: загрузка раунда, статистика между этапами, таймер-переход. Улучшена логика таймера — теперь он не теряет время между экранами;
- адаптивная верстка: интерфейс был адаптирован под различные разрешения экранов: от мобильных телефонов до мониторов 4K. Это обеспечило универсальность и соответствие техническому заданию.

На завершающем этапе разработки Максим сосредоточился на завершении функциональности, улучшении пользовательского опыта и подготовке продукта к финальной демонстрации:

- создание вспомогательного приложения для генерации координат: Максим разработал отдельный инструмент, позволяющий: загружать изображение; вручную выделять область с животным; автоматически генерировать JSON-структуру с координатами, совместимую с базой данных

проекта. Это значительно ускорило и упростило процесс наполнения уровней аналитиком и тимлидом, сократило вероятность ошибок и сделало масштабирование данных возможным без привлечения разработчиков;

– завершение стилизации и адаптивности: Максим завершил реализацию всех визуальных компонентов, включая: модальные окна; кнопки управления; индикаторы прогресса; шапку и футер; состояние загрузки (лоадер);

– также была окончательно выстроена адаптивная сетка для корректного отображения интерфейса от 320px до 4K;

– мультиязычная поддержка: реализована полноценная система локализации: поддержка переключения языков в интерфейсе; загрузка переводов из базы данных; возможность расширения списка языков без изменения логики приложения;

– оптимизация пользовательского опыта (UX): внедрен анимированный лоадер при загрузке данных и переходах между уровнями; улучшена логика состояния интерфейса: предотвращение преждевременных кликов, ожидание загрузки уровня до активации таймера; добавлены промежуточные экраны между режимами;

– рефакторинг кода и устранение багов: Максим провел глубокую ревизию кодовой базы: удалены дублирующиеся компоненты; упрощена структура; оптимизирован рендеринг; устранины баги, влияющие на отображение элементов и стабильность игрового процесса.

Максим реализовал весь пользовательский интерфейс, начиная от архитектурной заготовки и заканчивая финализированной адаптивной, стилизованной, мультиязычной системой. Его зона ответственности охватывала:

- проработку логики двух игровых режимов;
- точное взаимодействие с координатами;
- реализацию UX/UI согласно макетам;

- подключение к API и работе с данными;
- создание технической инфраструктуры для деплоя.

Кроме того, его вспомогательный инструмент (приложение для разметки координат) стал важнейшим звеном в наполнении уровней и работе аналитика.

Работа Максима позволила создать продукт, который одновременно технологичен, удобен, масштабируем и доступен для широкой аудитории. Он обеспечил полную реализацию клиентской части проекта, соответствующую современным требованиям к интерфейсу, производительности и гибкости.

9 Работа backend-разработчика

Влад занимал позицию backend-разработчика в проектной команде. Он отвечал за построение всей логики серверной части веб-приложения, разработку структуры базы данных, реализацию API для фронтенда, хранение и передачу изображений, настройку безопасности и поддержку мультиязычного контента. Его работа обеспечила техническую основу, на которой строилось взаимодействие между всеми компонентами системы.

На первом этапе разработки Владом были выполнены задачи:

– проектирование структуры базы данных: Влад разработал систему сущностей и связей между ними, включающую:

- 1) таблицу животных;
- 2) таблицы описаний и названий животных на разных языках;
- 3) таблицы уровней для первого и второго раундов;
- 4) структуру хранения координат, изображений и вариантов ответов;
- 5) таблицы интерфейсных текстов (UI) с поддержкой мультиязычности;

– построение логики уровней: для оптимизации архитектуры Влад применил наследование через абстрактные классы, позволившее избежать дублирования кода между первым и вторым игровыми раундами. Это решение улучшило расширяемость проекта;

– разработка системы выдачи уровней: он реализовал систему, при которой каждому пользователю выдается уникальный набор уровней без повторов;

– первичная работа с изображениями: был сделан шаг к оптимизации: вместо передачи двух изображений (исходное и размеченное) Влад предложил хранить координаты и отдавать их фронту, минимизируя нагрузку на хранилище.

На втором этапе разработки Владом были выполнены задачи:

– передача изображений через Base64: Влад разработал кастомный сериализатор изображений, позволивший: безопасно передавать фото через API; обрабатывать загрузку изображений на стороне сервера; сократить объем запросов и упростить интеграцию с фронтеном;

– оптимизация хранения данных: изменен подход — теперь каждый уровень хранит одну картинку и координаты животного, без лишнего дублирования. Это ускорило выдачу уровней и уменьшило объем хранимых данных;

– рефакторинг кода: Влад активно улучшал читаемость и надежность системы;

– подготовка к мультиязычности: проект начал активно переходить к поддержке нескольких языков. Влад спроектировал соответствующую структуру: для названий и описаний животных; для UI-текстов интерфейса.

На третьем этапе Влад сосредоточился на доработке сложных технических компонентов: локализация, динамическое построение уровней на разных языках, работа с текстами интерфейса, а также защита внутреннего API:

– полноценная реализация мультиязычности на сервере: Влад внедрил поддержку нескольких языков на уровне всей backend-системы. Это включало: хранение названий животных на разных языках (таблица `animal_name`); хранение описаний животных (таблица `animal_description`); создание интерфейсных текстов и их переводов (таблицы `ui_texts` и `multilingual_ui_texts`). Все данные теперь выдаются клиенту с учетом выбранного языка, что позволило фронтенду динамически адаптировать интерфейс без дублирования логики;

– умная логика создания уровней с учетом языка: Влад реализовал алгоритмы, которые: позволяют создавать уровни с животными независимо от языка — система сама сопоставляет нужные записи. Это решение упростило работу аналитика и тимлида при заполнении уровней, и сделало структуру базы масштабируемой;

– безопасность Swagger-документации: по требованию заказчика Влад реализовал ограничение доступа к Swagger UI (интерактивной документации API): он внедрил кастомный HTTP-фильтр, который требует авторизацию при попытке открыть Swagger; при этом основное приложение не перегружено полноценным Spring Security, что ускоряет запуск и упрощает поддержку.

Такое решение сохранило баланс между безопасностью и легкостью разработки.

Влад обеспечил всю серверную часть проекта и сформировал надежную техническую архитектуру, на которой держится игра. Его вклад включает:

- проектирование и реализацию БД;
- обработку изображений и метаданных;
- бизнес-логику уровней и сессий;
- настройку API и сериализации данных;
- поддержку мультиязычности и безопасности;
- оптимизацию производительности и читаемости кода.

Функциональность продукта, масштабируемость, точность выдачи уровней, стабильность загрузки и гибкость интерфейса — все это стало возможным благодаря работе Влада. Его технические решения позволили реализовать сложные задачи в сжатые сроки и сделали backend системы устойчивым, расширяемым и понятным для интеграции с frontend'ом.

ЗАКЛЮЧЕНИЕ

В результате разработки был создан адаптивный веб-продукт в формате интерактивной игры, полностью отражающей цели, обозначенные в техническом задании. Проект реализует ключевую идею заказчика — вовлечение пользователей в процесс распознавания животных с использованием механик геймификации и демонстрации работы ИИ.

Важно отметить, что программный продукт не просто выполняет функциональные требования, но и отвечает потребностям целевых аудиторий:

- любители природы получают эмоционально вовлекающий и познавательный опыт;
- ученые — инструмент для дальнейшего использования в научных целях;
- студенты и преподаватели — полезный образовательный инструмент, наглядно демонстрирующий принципы зоологического мониторинга и ИИ.

Таким образом, продукт достиг ценностного соответствия, то есть не просто решает технические задачи, но и выполняет образовательную и просветительскую функции, изначально заложенные в замысел проекта.

Проведенные тестирования показали, что программный продукт:

- работает стабильно на разных типах устройств и разрешениях экрана;
- корректно отображает все этапы игрового процесса и результаты;
- устойчив к повторным загрузкам и действиям пользователя вне сценария.

Несмотря на наличие технических ошибок на промежуточных этапах, все выявленные баги были оперативно устранены. Это говорит о высоком уровне внутренней коммуникации и ответственности команды.

Основной вывод: обнаруженные дефекты не носили критического характера и не повлияли на общую работоспособность продукта. Сложные

участки были реализованы эффективно и стablyно функционируют в финальной версии.

Проект обладает высоким потенциалом для масштабирования и улучшения. На основе проведенной работы можно выделить следующие направления развития:

- добавление новых раундов (например, распознавание следов животных или голосов);
- расширение базы животных, внедрение редких или региональных видов;
- интеграция с реальными системами мониторинга (партнерские научные проекты, национальные парки);
- внедрение реального ИИ-движка для распознавания изображений в реальном времени;
- реализация сохранения прогресса игры между сессиями пользователя;
- анимации, звуковое сопровождение, прогресс-бары;
- поддержка дополнительных языков для расширения международной аудитории;
- проведение онлайн-соревнований среди учащихся (геймифицированная олимпиада).

Разработка данного проекта стала не только успешной реализацией технического задания, но и примером того, как цифровые технологии, геймификация и наука могут быть объединены в одном продукте. Команда показала высокий уровень самоорганизации, гибкости и ориентации на результат.

Полученный продукт имеет устойчивую основу для масштабирования, потенциал для применения в образовании и популяризации науки, а также открывает перспективы для дальнейших технологических экспериментов с ИИ в гуманитарных и экологических проектах.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Figma: сайт для дизайна интерфейсов [Электронный ресурс]. – 2025. – URL: <https://www.figma.com/> (дата обращения: 27.04.2025).
2. UX Design в веб-разработке [Электронный ресурс]. – 2025. – URL: <https://uxdesign.cc/> (дата обращения: 27.04.2025).
3. React: официальный сайт [Электронный ресурс]. – 2025. – URL: <https://react.dev/> (дата обращения: 10.05.2025).
4. TypeScript: документация [Электронный ресурс]. – 2025. – URL: <https://www.typescriptlang.org/docs/> (дата обращения: 10.05.2025).
5. Java 21: официальная документация [Электронный ресурс]. – 2025. – URL: <https://docs.oracle.com/en/java/> (дата обращения: 12.05.2025).
6. Spring Framework: официальный сайт [Электронный ресурс]. – 2025. – URL: <https://spring.io/> (дата обращения: 12.05.2025).
7. PostgreSQL: документация [Электронный ресурс]. – 2025. – URL: <https://www.postgresql.org/docs/> (дата обращения: 11.05.2025).
8. JSON: структура и синтаксис [Электронный ресурс]. – 2025. – URL: <https://www.json.org/json-en.html> (дата обращения: 16.05.2025).
9. OpenAPI & Swagger: документация [Электронный ресурс]. – 2025. – URL: <https://swagger.io/docs/> (дата обращения: 20.05.2025).
10. Docker: официальная документация [Электронный ресурс]. – 2025. – URL: <https://docs.docker.com/> (дата обращения: 11.05.2025).

ПРИЛОЖЕНИЕ А

Основные страницы сайта

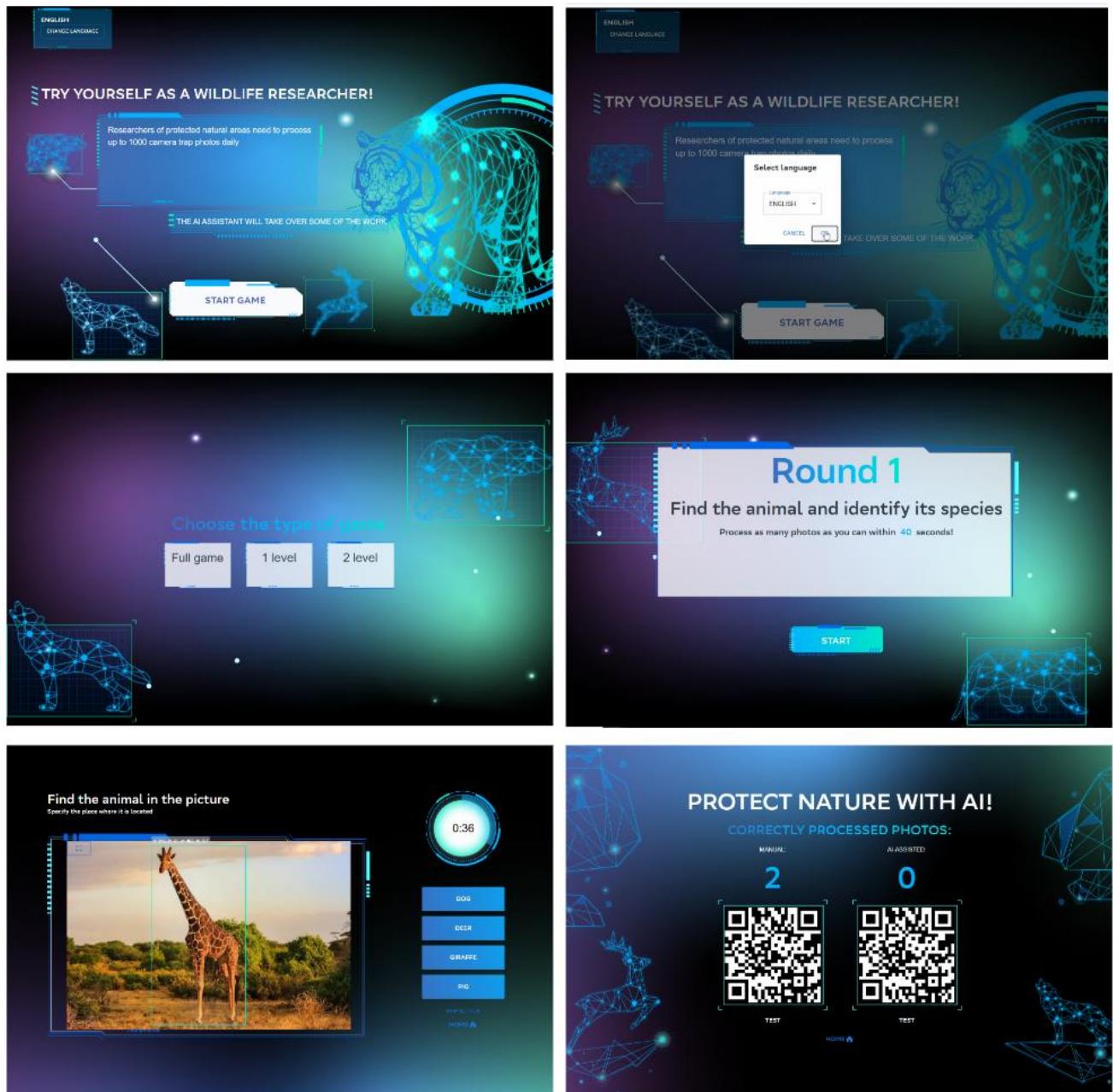


Рисунок 1 - Основные страницы игры