**ДАТА УТВЕРЖДЕНИЯ**

# I ПАСПОРТ ПРОЕКТА

| Наименование проекта | *Определение эвристик по поиску интересных моментов в шахматной партии* |
| --- | --- |
| Шифр проекта (команды) | *Мадагаскам* |
| Заказчик проекта | *Компания Friflex* |
| Руководитель темы от МАИ | *Булакина Мария Борисовна* |
| Целевая аудитория результата проекта (кто потребитель результата проекта) | *Шахматные любители, шахматные клубы, шахматные видеоблогеры, платформы для онлайн-игр и турниров.* |
| Длительность проекта (даты начала и окончания) |  |
| РОЛИ В ПРОЕКТЕ: | ФИО |
| Teamlead | *Шаталов Максим Алексеевич* |
| Python Backend | *Авраменко Денис Александрович* |
| ML engineer | *Агафонов Андрей Сергеевич* |
| Python Backend | *Белоусов Донат Александрович* |
| Аналитик | *Голосов Георгий Сергеевич* |
| ML engineer | *Ивченко Матвей Сергеевич* |
| Frontend | *Лапенко Карина Александровна* |
| Python Backend | *Мельцова Вероника Алексеевна* |
| ML engineer | *Пономарев Артём Андреевич* |
| Дата создания первой версии паспорта проекта | *01.12.2024* |

Ссылки на ресурсы проекта

|  |  |
| --- | --- |
| Ссылка на GitHub | 1. <https://github.com/Maxishoo/MadagascamChess> |

# II ОПИСАНИЕ ПРОЕКТА

|  |  |
| --- | --- |
| Образ результата | *IT-продукт* |
| Цель проекта | *Разработать сервис для автоматической обработки PGN шахматной партии, размечающий её на наиболее интересные отрезки* |
| Задачи проекта |  |
| 1 | *Разработать аналитическую модель определения интересного момента* |
| 2 | *Собрать и разметить данные шахматных партий* |
| 3 | *Обучить нейронную модель для определения интересных моментов* |
| 4 | *Обучить нейронную модель для определения интересных моментов* |
| 5 | *Обучить нейронную модель для определения интересных моментов* |
| 6 | *Разработать интерфейс, визуализирующий работу сервиса* |
| Результат проекта (конкретный итоговый результат проекта) | *Автоматизированная система, которая на основе шахматных партий в формате PGN выделяет ключевые моменты, обрабатывает видеозаписи партий с использованием алгоритмов вырезки, склейки, наложения эффектов и нейросетевых преобразований, создавая привлекательный видеоконтент. Система полностью автоматизирована и реализована в виде программного кода, готового к интеграции в платформу idChess.* |
| Ограничения и допущения, которые имеют или могут оказать существенное влияние на результат проекта | *Основные ограничения проекта связаны с качеством входных данных (PGN-файлов и видеозаписей), вычислительными ресурсами для обработки видео и нейросетевых преобразований, а также точностью алгоритмов определения ключевых моментов, зависящей от обучающего набора.* |
| Необходимые ресурсы для выполнения проекта (компетенции исполнителей, материальные ресурсы и др.) | *Потребуются специалисты по шахматам, машинному обучению, обработке видео и разработке программного обеспечения, а также вычислительные ресурсы для тренировки нейросетей и обработки видео, качественные PGN-файлы и видеозаписи партий, программные инструменты для разработки и обработки данных.* |
| Риски проекта | *Недостаточное качество PGN-файлов и видеозаписей, что может снизить точность анализа.*  *Возможные ошибки нейросетей из-за ограниченности обучающего набора.*  *Нехватка вычислительных ресурсов для обработки больших объемов данных.* |

# 

# III КОМАНДА ПРОЕКТА

|  |  |  |  |
| --- | --- | --- | --- |
| **ФИО** | **Роль** | **Компетенция** | **Задача проекта** |
| Шаталов Максим Алексеевич | Teamlead | 1. Python 2. OpenCV, Tenserflow 3. Scikit-learn 4. Pandas, Numpy |  |
| Авраменко Денис Александрович | Python Backend | 1. Python, Django, FastAPI 2. PostgreSQL 3. RESTful API |  |
| Агафонов Андрей Сергеевич | ML engineer | 1. Python 2. Tenserflow 3. Scikit-learn 4. Pandas, Numpy |  |
| Белоусов Донат Александрович | Python Backend | 1. Python, Flask, FastAPI 2. PostgreSQL |  |
| Голосов Георгий Сергеевич | Аналитик | 1. Python 2. Pandas 3. Excel 4. Power BI |  |
| Ивченко Матвей Сергеевич | ML engineer | 1. Python 2. Keras 3. Scikit-learn 4. Pandas, Numpy |  |
| Лапенко Карина Александровна | Frontend | 1. JavaScript 2. HTML/CSS 3. React, Angular |  |
| Мельцова Вероника Алексеевна | Python Backend | 1. Python, Django, Flask, FastAPI 2. MongoDB |  |
| Пономарев Артём Андреевич | ML engineer | 1. Python 2. Pytorch 3. Scikit-learn 4. Pandas, Numpy |  |

# 

# IV ЗАДАЧИ ПРОЕКТА (ОЦЕНКА ПО ВРЕМЕНИ)

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Название задачи** | **Подзадачи** | **Время**  **(в часах)** |
| 1 | Настройка проекта в среде разработки | 1. Установка и настройка Spring Boot и ReactJS проектов 2. Интеграция с PostgreSQL | 6 |
| 2 | Разработка бекенда на Spring Boot | 1. Создание основных классов 2. Настройка подключения к PostgreSQL 3. Настройка REST API | 30 |
| 3 | Разработка системы авторизации пользователей | 1. Реализация регистрации 2. Реализация входа 3. Управление пользователями | 30 |
| 4 | Разработка системы бронирования | 1. Создание API для бронирования аудиторий и инвентаризации 2. Обработка запросов 3. Валидация данных | 40 |
| 5 | Разработка вспомогательных сервисов | 1. Реализация сервисов для уведомлений и других вспомогательных функций | 20 |
| 6 | Разработка модели данных в PostgreSQL | 1. Определение схемы базы данных 2. Создание таблиц для бронирования аудиторий и инвентаризации оборудования | 12 |
| 7 | Разработка frontend на ReactJS | Создание компонентов для:   1. Окна авторизации 2. Главной страницы 3. Создания и изменения бронирования 4. Панели администратора | 50 |
| 8 | Написание стилей | 1. Написание CSS для стилизации компонентов 2. Обеспечение единообразного дизайна приложения | 25 |
| 9 | Интеграция бекенда и фронтенда | 1. Обеспечение взаимодействия между ReactJS и Spring Boot 2. Тестирование API | 25 |
| 10 | Разработка системы уведомлений | 1. Настройка уведомлений для подтверждения бронирования и напоминаний | 10 |
| 11 | Разработка панели администратора | 1. Реализовать возможность редактирования доступных действий для роли 2. Разработка сервиса для логирования 3. Вывод логов приложения с фильтрацией и поиском 4. Вывод статистической информации о бронированиях | 30 |
| 12 | Разработка алгоритма оптимизации распределения аудиторий | 1. Определение математической модели 2. Выбор метода оптимизации для реализации алгоритма 3. Тестирование алгоритма для проверки корректности 4. Разработка микросервиса | 100 |
| 13 | Разработка микросервиса для парсинга расписания | 1. Реализовать сервис для парсинга расписания 2. Реализовать сервис для обогащения и группировки элементов расписания | 30 |
| 14 | Тестирование приложения и подготовка протокола | 1. Написание и проведение тестов 2. Составление документации 3. Подготовка протокола тестирования | 15 |
| 15 | Составление документации по использованию приложения | 1. Написание руководства пользователя 2. Описание функционала 3. Создание презентации | 10 |
| 16 | Развертывание приложения | 1. Настройка сервера 2. Размещение бекенда и фронтенда 3. Проверка работоспособности | 15 |
| **ИТОГО ПЛАНИРУЕМОЕ ВРЕМЯ НА ПРОЕКТ :** | | | **448** |