### **План действий и распределение ресурсов**

команды “Las Teteras Desesperadas”

| **Эпики и задачи** \*Ответственный | **О\*** | **Дата** |
| --- | --- | --- |
| **1. Инициация проекта** |  | 28-29.11 |
| 1.1. Встреча с заказчиком |  | 28.11 |
| - Встретиться с заказчиком для ознакомления с целями и задачами проекта. | Все | 28.11 |
| - Уточнить непонятные моменты в требованиях и задачах. | Все | 28.11 |
| 1.2. Формирование проектной команды |  | 28.11 |
| - Определить необходимые роли в проекте:  проект-менеджер (PM), дата-инженер (DE), дата-аналитик (DA), ML-инженер (MLE), QA-инженер (QAE), технический писатель (TW). | Все | 28.11 |
| - Назначить ответственных на каждую роль:  PM - Загальский Игорь  DE - Загальский Игорь, Майнгерт Владимир  DA - Загальский Игорь, Санникова Юлия, Антропова Наталия  MLE - Загальский Игорь, Санникова Юлия  QAE - Антропова Наталия  TW - Мухачев Иван, Аванесян Тачат | Все | 28.11 |
| - Определить коммуникационные каналы в команде:  чат в Telegram, репозиторий в Git. | Все | 28.11 |
| 1.3. Подготовка устава проекта |  | 29.11 |
| - Сформировать документ с описанием целей, ожидаемых результатов и критериев успеха проекта. | PM | 29.11 |
| 1.4. Определение необходимых ресурсов |  | 29.11 |
| - Провести анализ необходимых инструментов и технологий для разработки рекомендательной системы. | PM | 29.11 |
| - Рассчитать временные ресурсы для каждой задачи:  см. Таймлайн проекта. | PM | 29.11 |
| - Рассчитать трудовые ресурсы для каждой задачи:  назначить ответственных в за каждую задачу в колонке \*О. | PM | 29.11 |
| - Рассчитать финансовые затраты для каждой задачи:  при нулевом бюджете финансовые затраты также нулевые. | PM | 29.11 |
| **2. Планирование** |  | 30.11 - 3.12 |
| 2.1. Разработка плана действий |  | 30.11 |
| - Составить детальный список на время проведения практической точки, включающий описание этапов реализации проекта, распределение задач и ресурсов, оценку времени, необходимого для каждой задачи и прочую важную информацию, необходимую для успешной реализации. | PM | 30.11 |
| 2.2. Составление таймлайна проекта и реестра рисков |  | 1.12 |
| - Построить наглядное отображение плана действий с разбивкой задач по дням проекта. | PM | 1.12 |
| - Произвести анализ потенциальных рисков, связанных с реализацией проекта. | PM | 1.12 |
| - Предложить стратегию управления рисками и их минимизации. | PM | 1.12 |
| 2.3. Подготовка черновика решения |  | 2-3.12 |
| - Сформировать логичное, чётко структурированное решение для чек поинта с заказчиком, имеющее последовательность шагов для реализации проекта. | PM | 2-3.12 |
| **3. Сбор и обработка данных** |  | 29.11 - 8.12 |
| 3.1. Сбор данных |  | 29.11 - 8.12 |
| - Настроить парсинг больших данных через API сервис UniGoTo. | DE | 29.11 |
| - Найти и удалить строки с ошибочными значениями в столбцах типа Integer: 'university', 'faculty', 'graduation', 'country\_id', 'city\_id'. | DE | 30.11 |
| - Загрузить данные в один большой датафрейм:  - загрузка спарсенных данных в датафрейм Pandas;  - создание вспомогательных таблиц c id ВУЗов, факультетов, стран и городов; - удаление строк, где значения превышают пороговые значения;  - удаление строк, где названия ВУЗов указаны не на русском языке, а незаполненные - оставлены. | DE | 30.11 |
| - Осуществить парсинг данных пользователей:  батчами по 250 страниц \* 999 записей на странице. | DE | 30.11 - 8.12 |
| 3.2. Обработка данных |  | 1-8.12 |
| - Провести предварительную фильтрацию данных:  - удаление дубликатов;  - удаление строк, в которых количество заполненных столбцов интересов < 3;  - удаление строк, содержащих рекламные слова в столбцах интересов;  - удаление строк, где значения столбцов с числовыми значениями = 0;  - преобразование типов данных столбцов с числовыми значениями преобразованы в int16 и int32. | DA | 1-2.12 |
| - Очистить интересы от лишних символов и шумов:  - замена пробелами всех знаков, кроме рус., англ. букв и пробелов;  - приведение к нижнему регистру;  - удаление сочетаний более 3 подряд идущих букв;  - удаление повторяющихся слов;  - удаление пробелов в начале и конце строки;  - замена подряд идущих пробелов на один;  - удаление строк с 'NaN' во всех 5 столбцах интересов. | DA | 3-4.12 |
| - Отфильтровать вспомогательные таблицы:  - удаление дубликатов;  - сортировка по возрастанию первых столбцов;  - фильтрация названий ВУЗов и факультетов;  - удаление символов переноса строки;  - создание файла соответствия города и региона. | DA | 5-6.12 |
| - Провести предобработку интересов:  - токенизация текста на отдельные слова;  - удаление стоп-слов для русского и английского языков;  - лемматизация слов. | DA | 7-8.12 |
| **4. Создание рекомендательной системы** |  | 9-10.12 |
| 4.1. Выбор и обучение моделей |  | 9-10.12 |
| - Выбрать подходящие модели для рекомендации:  - на основе косинусного сходства с разреженной матрицей TF-IDF - используется для поиска сходства между двумя документами (выбрана);  - NearestNeighbors - baseline (использовался заказчиком);  - LightFM (рекомендована Татьяной Булгаковой). | MLE | 9.12 |
| - Модель на основе косинусного сходства:  - создание разреженной матрицы TF-IDF для преобразования каждой строки интересов пользователей в вектор числовых значений, отражающий важность слов в каждой строке относительно всего набора данных (векторизация интересов);  - обучение на размеченных данных, а следовательно и разделение данных на обучающую и тестовую выборки не требуется, так как эта модель для рекомендации, а не для обучения с учителем;  - вычисление косинусного сходства между обучающей выборкой и матрицей TF-IDF параллельным способом;  - сохранение рекомендаций вычисления косинусного сходства;  - при оценке модели рекомендаций без использования обучения с учителем, нет простого способа оценить качество модели с использованием стандартных метрик. | MLE | 9-10.12 |
| 4.2. Реализация ранжирования результатов по различным параметрам |  | 10.12 |
| - Указывать строку ввода города при вводе данных пользователем. | MLE | 11.12 |
| - При выводе предоставлять отдельный ранжированный список с ограничением мест обучения по городу или региону. | MLE | 11.12 |
| - Предусмотреть возможность сделать ранжирование по факультетам. | MLE | 11.12 |
| - Перевести id в имена ВУЗов и факультетов после нахождения рекомендаций. | MLE | 11.12 |
| **5. Интеграция и тестирование** |  | 11-13.12 |
| 5.1. Интеграция компонентов решения |  | 11-12.12 |
| - Создать Git репозиторий с хорошо организованной структурой:  - в корне лежат разные настройки и [README.md](https://readme.md);  - блокноты и скрипты лежат в папке src/;  - исходные спарсенные данные лежат в папке raw\_data/;  - промежуточные и финальные данные лежат в папке data/;  - модели лежат в папке models/;  - рекомендация проекта лежит в папке project\_docs/. | MLE | 11.12 |
| - Связать все компоненты решения в единое целое:  - Pipeline:  1. Парсинг данных с API  script: src/parser.py  out: raw\_data/\*.zip  2. Проверка столбцов Integer  script: src/int\_checker.py  in: raw\_data/\*.zip  out: raw\_data/\*.zip  3. Загрузка данных  script: src/loader.py  in: raw\_data/\*.zip  out: data/combined\_data.csv, cities.csv, countries.csv, faculties.csv, universities.csv  4. Фильтрация вспомогательных таблиц  script: src/aux\_filter.py  in: data/cities.zip, countries.zip, faculties.zip, universities.zip, csv  out: data/cities\_filtered.csv, countries\_filtered.csv, faculties\_filtered.csv, universities\_filtered.csv  5. Создание таблицы регионов  script: src/regions.py  in: data/cities\_filtered.zip, regions.zip, csv  out: data/cities\_regions.zip, csv  6. Предварительная фильтрация  script: src/pre\_filter.py  in: data/combined\_data.zip, csv  out: data/prefiltered\_data.csv  7. Очистка интересов  script: src/noise\_filter.py  in: data/prefiltered\_data.zip, csv  out: data/filtered\_data.csv  8. Предобработка интересов  script: src/preprocessor.py  in: data/filtered\_data.zip, csv  out: data/preprocessed\_data.csv  \*. Модель на основе косинусного сходства (не нужно)  script: src/cosine\_sims.py  in: data/preprocessed\_data.zip, csv  out: recommendations/\*.csv  9. Рекомендатор  script: src/recommendator.py  in: data/preprocessed\_data.zip, cities\_regions.zip, cities\_filtered.zip, countries\_filtered.zip, faculties\_filtered.zip, universities\_filtered.zip, csv  out: output/user\_data.csv  - fast API и streamlit (нужны файлы:  - src/recommendator.py - data/preprocessed\_data.zip  - data/cities\_regions.zip  - data/cities\_filtered.zip  - data/countries\_filtered.zip  - data/faculties\_filtered.zip  - data/universities\_filtered.zip | MLE | 11-12.12 |
| 5.2. Тестирование решения на совместимость и работоспособность |  | 12.12 |
| - Провести тестирование решения на совместимость и работоспособность:  на платформе streamlit. | QAE | 12.12 |
| - Зарегистрировать и исправить обнаруженные ошибки. | QAE | 12.12 |
| 5.3. Доработка функционала |  | 12-13.12 |
| - Внести изменения в функционал решения по результатам тестирования. | MLE | 12-13.12 |
| **6. Разработка документации** |  | 3-13.12 |
| 6.1. Создание технической документации |  | 3-13.12 |
| - Документировать процессы обработки данных, обучения модели и использования решения. | TW | 3-13.12 |
| - Подготовить инструкции по переобучению модели. | TW | 12-13.12 |
| 6.2. Документирование кода |  | 7-12.12 |
| - Написание подробных комментариев в коде рекомендательной системы. | TW | 7-12.12 |
| **7. Завершение проекта** |  | 12-15.12 |
| 7.1. Подготовка отчёта о реализации проекта |  | 12-14.12 |
| - Заполнить созданный ранее Git репозиторий для подготовки чистовика решения. | MLE | 12-13.12 |
| - Предоставить чистовик решения и документацию заказчику. | Все | 13.12 |
| - Создать презентацию о реализации проекта и достижении поставленных целей. | QAE | 11-14.12 |
| 7.2. Подведение итогов |  | 15.12 |
| - Защитить проект, опираясь на презентацию и чистовик решения. | Все | 15.12 |
| - Подвести итоги проекта. | Все | 15.12 |
| - Выделить достижения команды. | Все | 15.12 |
| **8. Дополнительные задачи (при наличии времени)** |  | 12-13.12 |
| 8.1. Реализация решения в виде Docker-контейнера |  | 12.12 |
| - Создать контейнер для решения с использованием Docker. | MLE | 12.12 |
| 8.2. Реализация взаимодействия через API |  | 12.12 |
| - Добавить функционал для взаимодействия через API, предпочтительно в виде GRPC. | MLE | 12.12 |
| - Описать схему взаимодействия. | TW | 12.12 |