

Детальные задачи

📌 TASK-001: Детализация функциональных требований (5 SP)

Исполнитель: Ивашнев Семен

Кандидаты: Ивашнев, Сергеев

Описание:

Создать полную спецификацию функциональных требований системы на основе описания проекта. Документ должен описывать все функции MVP и полной версии системы.

Что конкретно делать:

1. Создать документ "Функциональные требования Speak2MD" в формате Word
2. Описать требования к загрузке аудио:
 - Поддерживаемые форматы (mp3, m4a, wav)
 - Ограничения (50 МБ / 120 минут для free тарифа)
 - Валидация файлов
3. Описать требования к записи аудио через микрофон:
 - API браузера MediaRecorder
 - Форматы записи
 - UI элементы управления
4. Детализировать процесс обработки:
 - VAD (Voice Activity Detection) через silero-vad
 - ASR через NVIDIA Parakeet-TDT-0.6B
 - Постпроцессинг через Qwen3-1.5B
5. Требования к выходному формату:
 - Структура Markdown файла
 - Таймкоды
 - Форматирование заголовков и списков

Критерии приемки:

- Документ содержит минимум 15 функциональных требований
- Каждое требование имеет ID, приоритет, описание
- Указаны нефункциональные требования (производительность, безопасность)

- Документ согласован с РМ

Результат: Файл "Functional_Requirements_Speak2MD.docx"

TASK-002: Создание пользовательских историй (3 SP)

Исполнитель: Азиз Кодиров

Кандидаты: Азиз, Ивашнев

Описание:

Написать User Stories для всех ключевых сценариев использования системы в формате "Как [роль], я хочу [действие], чтобы [результат]".

Что конкретно делать:

1. Создать файл "User_Stories_Speak2MD.md"
2. Написать минимум 10 пользовательских историй:
 - US-001: Загрузка аудио файла
 - US-002: Запись аудио с микрофона
 - US-003: Отслеживание прогресса обработки
 - US-004: Скачивание результата в MD
 - US-005: Просмотр результата в браузере
 - US-006: Регистрация пользователя
 - US-007: Выбор тарифного плана
 - US-008: Поиск по транскриптам
 - US-009: Создание ссылки для шеринга
 - US-010: Управление квотами
3. Для каждой истории указать:
 - Acceptance Criteria (критерии приемки)
 - Priority (High/Medium/Low)
 - Story Points

Критерии приемки:

- Минимум 10 user stories покрывают основной функционал
- Формат соответствует Agile стандартам
- Есть критерии приемки для каждой истории
- Истории проверены с командой

Результат: Файл "User_Stories_Speak2MD.md" в репозитории

TASK-003: Составление сметы проекта (3 SP)

Исполнитель: Сергеев Александр

Кандидаты: Сергеев

Описание:

Рассчитать полную смету проекта в предоставленном шаблоне Excel с учетом всех статей расходов.

Что конкретно делать:

1. Открыть шаблон сметы от преподавателя
2. Заполнить раздел "Трудозатраты":
 - PM: 80 часов × 3000 руб/час
 - Backend Dev (2 чел): 160 часов × 2500 руб/час
 - Frontend Dev: 60 часов × 2000 руб/час
 - QA: 40 часов × 1800 руб/час
3. Раздел "Инфраструктура":
 - VPS сервер: 3000 руб/мес × 2 мес
 - PostgreSQL hosting: 2000 руб/мес × 2 мес
 - Redis Cloud: 1500 руб/мес × 2 мес
 - Домен: 500 руб/год
4. Раздел "Лицензии и сервисы":
 - GitHub Team: \$4/user/month × 4 users × 2 months
 - YouTrack: бесплатно (до 10 пользователей)
5. Накладные расходы: 10% от ФОТ
6. Резерв на риски: 15% от общей суммы

Критерии приемки:

- Все разделы шаблона заполнены
- Итоговая сумма с обоснованием
- Добавлен график платежей
- Смета утверждена заказчиком

Результат: Файл "Смета_Speak2MD_v1.xlsx"

TASK-004: Разработка плана в MS Project (5 SP)

Исполнитель: Сергеев Александр

Кандидаты: Сергеев, Саватин

Описание:

Создать детальный план проекта в MS Project с диаграммой Ганта, ресурсами и вехами.

Что конкретно делать:

1. Создать новый проект в MS Project
2. Настроить календарь проекта:
 - Рабочие дни: Пн-Пт
 - Рабочие часы: 18:00-22:00 (part-time)
 - Выходные учесть
3. Добавить все задачи из 4 спринтов с иерархией:
 - Спринт 1 (родительская задача)
 - TASK-001 ... TASK-007 (подзадачи)
 - Аналогично для остальных спринтов
4. Назначить ресурсы (членов команды) на задачи
5. Установить зависимости между задачами
6. Добавить вехи (milestones):
 - Конец каждого спрингта
 - Контрольная точка 2 (31.10)
7. Настроить базовый план (baseline)
8. Добавить критический путь

Критерии приемки:

- План содержит все 29 задач проекта
- Ресурсы не перегружены (макс 100% загрузки)
- Критический путь визуализирован
- Экспорт в PDF для презентации

Результат: Файлы "Speak2MD_Project.mpp" и "Speak2MD_Gantt.pdf"



TASK-005: Создание паспорта проекта (5 SP)

Исполнитель: Саватин Виталий

Кандидаты: Саватин, Сергеев

Описание:

Заполнить официальный паспорт проекта по предоставленному шаблону с полной информацией о проекте.

Что конкретно делать:

1. Использовать шаблон паспорта от преподавателя
2. Заполнить раздел "Общие сведения":
 - Название: Speak2MD
 - Заказчик: [ФИО преподавателя]
 - Исполнитель: Команда группы 3371
 - Сроки: 07.10.2025 - 19.12.2025
3. Раздел "Цели и задачи":
 - Цель: Создание веб-сервиса автоматической транскрипции
 - Задачи: разбить на 9 этапов курсовой
4. Раздел "Результаты проекта":
 - MVP версия системы
 - Техническая документация
 - Исходный код в Git
5. Раздел "Ограничения и допущения":
 - Бюджет: согласно смете
 - Технологии: Python, FastAPI, PostgreSQL
 - Сроки: жесткие по контрольным точкам
6. Раздел "Риски" (минимум 5):
 - Технические сложности с ML моделями
 - Нехватка времени команды
 - И другие
7. Раздел "Команда проекта" с ролями

Критерии приемки:

- Все разделы шаблона заполнены
- Документ подписан РМ
- Согласован с заказчиком
- Версия 1.0 зафиксирована

Результат: Файл "Паспорт_проекта_Speak2MD_v1.docx"



TASK-006: Определение критериев приемки (3 SP)

Исполнитель: Саватин Виталий

Кандидаты: Саватин, Азиз

Описание:

Разработать четкие критерии приемки для каждого этапа проекта и финальной системы.

Что конкретно делать:

1. Создать документ "Acceptance_Criteria_Speak2MD.docx"
2. Описать критерии для MVP (Контрольная точка 2):
 - Загрузка файла работает для mp3/m4a/wav
 - Ограничение 50МБ корректно валидируется
 - Прогресс обработки отображается в реальном времени
 - Результат скачивается в формате .md
 - Система выдерживает 5 параллельных запросов
3. Критерии для полной версии:
 - Регистрация/авторизация работает
 - Тарифные планы корректно ограничивают функционал
 - История транскриптов сохраняется
 - Поиск работает по всем транскриптам
4. Технические критерии:
 - Unit test coverage > 60%
 - Все API endpoints задокументированы
 - Deployment автоматизирован через CI/CD
5. Критерии для документации:
 - README с инструкцией по запуску
 - API документация в Swagger
 - Руководство пользователя с скриншотами

Критерии приемки:

- Минимум 20 измеримых критериев
- Критерии разбиты по категориям
- Каждый критерий можно проверить да/нет
- Согласовано с командой

Результат: Файл "Acceptance_Criteria_Speak2MD.docx"

📌 TASK-007: Описание API контрактов (5 SP)

Исполнитель: Ивашнев Семен

Кандидаты: Ивашнев, Сергеев

Описание:

Создать полную спецификацию API в формате OpenAPI/Swagger для всех endpoints системы.

Что конкретно делать:

1. Создать файл "api_specification.yaml" в формате OpenAPI 3.0
2. Описать базовые endpoints MVP:

```
POST /api/upload- Request: multipart/form-data с аудио файлом-
Response: {"job_id": "uuid", "status": "processing"}GET
/api/status/{job_id}- Response: {"status": "processing|ready|error",
"progress": 0-100}GET /api/result/{job_id}- Response: файл result.md
или {"error": "message"}
```

3. Endpoints для WebSocket:

```
WS /api/ws/{job_id}- События: progress, status_change, completed,
error
```

4. Endpoints авторизации:

```
POST /api/auth/registerPOST /api/auth/login POST
/api/auth/refreshGET /api/me
```

5. Endpoints для работы с транскриптами:

```
GET /api/transcriptsGET /api/transcripts/{id}POST
/api/transcripts/{id}/tagsDELETE /api/transcripts/{id}
```

6. Для каждого endpoint указать:

- HTTP метод и путь
- Параметры (path, query, body)
- Схемы request/response (JSON Schema)
- Коды ответов (200, 400, 401, 404, 500)
- Примеры запросов/ответов

Критерии приемки:

- Описаны все 15+ endpoints системы
- Файл валидный OpenAPI 3.0
- Можно импортировать в Postman

-  Swagger UI корректно отображает

Результат: Файлы "api_specification.yaml" и "API_Documentation.md"