Тестовое задание на позицию «Системный аналитик»

**Задание 1**

Необходимо разработать мобильное приложение для iOS которое позволяет пользователю подключить в приложение одного или нескольких виртуальных ассистентов (далее ВА). После подключения пользователь может общаться с ВА посредством голосового или текстового ввода. Одновременно общаться можно только с одним ВА.

Распознавание смысла, формирование ответа и логика работы ВА осуществляется в отдельном облачном сервисе (DP). Сервис DP получает на вход текст и выдает на выходе текст. Для взаимодействия с ВА предусмотрено API. Каждый ВА обладает уникальным uuid идентификатором.

Распознавание (ASR) и синтез речи (TTS) осуществляется в отдельных облачных сервисах. Сервис распознавания речи получает на вход аудио и выдает на выходе текст. Синтез речи производит обратную операцию. Для взаимодействия с каждым сервисом предусмотрено соответствующее API.

1) Сформируй use case диаграмму.  
2) Сформируй диаграмму состояний для use case описывающего подключение ВА в приложение.  
3) Распиши (текстом) подробно любой другой use case.

**Задание 2**

Представь, что мы планируем начать разработку приложения, которое состоит из различных микросервисов. Микросервисы взаимодействует между собой и отвечают за определенную функциональность, например, сервис авторизации, сервис логирования, сервис уведомлений и др. Для приложения необходимо начать готовить документацию.

1) Опиши, как бы ты структурировала документацию в общем.  
2) Опиши, какая информация/разделы (с описаниями, схемами, таблицами и прочими артефактами) должна быть в документации каждого микросервиса. Для каждого раздела поясни, почему ты считаешь, что данная информация необходима.

Не забудь учесть ряд важных моментов:

* Документация будет использоваться разработчиком во время разработки микросервисов и должна минимизировать (или вообще искоренить) вопросы у разработчика.
* Документация будет использоваться тобой (системным аналитиком) при последующей доработке архитектуры приложения, а также проработке новых функциональных возможностей.
* Документация должна быть максимально содержательной для погружения в работу приложения при появлении нового разработчика в компании.

**Задание 3**

Есть веб-виджет, который устанавливается на сайт и позволяет пользователям общаться с виртуальным ассистентом. Для разработки виджета используется библиотека chatKit – <https://github.com/sovaai/chatKit-lib>. Пример виджета, где используется библиотека chatKit с подключённым модулем DL – <https://dev.sova.ai/chat-kit-demo/>

Конфигурация отвечающая за отображение/скрытие элементов виджета сохраняется в локальном хранилище, которое реализовано с помощью бибилиотеки Storeon. В Storeon есть несколько различных модулей. Мы остановимся на модуле Management – <https://github.com/sovaai/chatKit-lib/blob/master/src/store/modules/managment.ts>. Management экспортируется из вне библиотеки – <https://github.com/sovaai/chatKit-lib/blob/master/src/index.tsx>. Дефолтные значения, которые хранится под этим ключом – <https://github.com/sovaai/chatKit-lib/blob/master/src/store/states/defaultStates.ts>. Описание (типизация) модуля –<https://github.com/sovaai/chatKit-lib/blob/master/src/store/%40types/storeonModules.d.ts>.

1) Подготовь документацию для Storeon модуля Management

**Срок выполнения задания 7 календарных дней с момента отправки.**