**Бизнес-требования**

**О заказчике:**

**Амбассадоры Практикума**

**Общее описание задачи:** Мы хотим создать инфраструктуру управления сообществом, которая приносит прогнозируемый поток маркетинговых привлечений в месяц.

На стороне менеджера сообщества, это CRM система с базой данных, набором функций и селекторами.   
Амбассадорское сообщество доказало свою бизнес состоятельность и задача, которая перед нами стоит - масштабировать сообщество и соответственно эффект от него.

(по состоянию на январь 2024 года мы сопровождаем уже около 150 амбассадоров.

Амбассадорские продажи составляют 3% от общего объема букингс Практикума. Ростом сообщества, мы хотим и увеличить эту долю.

Масштабирования сообщества, с которым мы уже сталкиваемся, увеличивает операционную нагрузку на менеджера этого сообщества. Нередко операционное или координационное обслуживание каждого амбассадора стоит комьюнити-менеджеру почти всего своего рабочего времени, хотя запросы от амбассадоров типичны и не требуют интеллектуального ресурса для ответа. Мы хотим, чтобы у нас появилась система, которая возьмет на себя координационные задачи, такие как ответы на частые вопросы, онбординг новых амбассадоров и другие.

**Рефренс: –**

**Целевая аудитория создаваемого сервиса:** выпускники Практикума ставшие амбассадорами бренда.

**Ожидаемый результат – MVP CRM-системы:**

1. **Внутренняя CRM система** - то, что видим и с чем работаем мы.  
   Пространство для комьюнити менеджера сообщества амбассадоров, в котором можно получать уведомления, делать рассылки и смотреть аналитику.
2. Технических чат-бот, который сообщает о действиях пользователей.  
   Основной функционал:  
   - Получение уведомлений о действиях из бота.  
   - Рассылка в личные сообщения (необходимая функция). Функционал рассылки по всем зарегистрированным в боте пользователям с возможность написать сразу всем зарегестрировавшимся в боте и выбором группы пользователей.

Чат-бот уже находится в разработке на финальной стадии!

**Основной функционал CRM системы:**

1. **Хранение данных с возможность редактирования и работы с ними (настроенная интеграция с Я Формами)**

* Вся информация по амбассадорам из формы
* Промокоды
* Контент
* Программа лояльности
* Отправка мерча
* Бюджет на мерч

Пример архитектуры базы данных - <https://docs.google.com/spreadsheets/d/1oEoYd03f-U0fhl_SrXxI0BEEuh35e4Ydz7WZcLFQYQg/edit?usp=sharing>

(сделан по подобию существующей системы хранения и работы с амбассадорами с небольшими изменениями).

**2. Аналитика и статистика**

* Что чаще всего спрашивают в боте.
* Список зарегистрировавшихся в боте (список).
* Итоги амбассадорской деятельности: сколько раз использовал бот, завершил ли онбординг, посмотреть срок нахождения амбассадора в сообществе (отсчитывается с момента добавления человека в бот), процент читаемости сообщений (получать уведомление, если амбассадор не прочитал 2 последних сообщения через бот).
* Получение уведомлений о действиях из бота.

**4. Рассылка в личные сообщения (лично и массово)**.

Функционал рассылки по всем зарегистрированным в боте пользователям с возможность написать сразу всем зарегистрировавшимся в боте и выбором группы пользователей.

*\*Идеальный вариант - возможность сегментировать запрос (только аналитикам, только мальчикам) такая информация у нас сейчас хранится в* [*трекере*](https://st.yandex-team.ru/agile/board/60716)*.*

Возможность отложенных массовых личных сообщений.

**На будущее (план дальнейшего развития может быть таким, по мнению заказчика):**

Уже существующий чат-бот и CRM система работаю в синхроне.

**Функциональные требования**

**Обязательные требования для работы frontend-разработчиков**Использовать стек технологий для работы React, TypeScript, Redut Toolkit, React hook form (для работы с формами), yup (для валидации), Material-UI (MUI для них может быть сложным, но в идеале кастомизировать его). Сверстать и подготовить верстку на основе результата работ команды дизайнеров (адаптив не обязателен, можно ориентироваться только на desktop). Интегрировать фронт-часть сервиса с командой backend разработчиков.

**Обязательные требования для работы backend-разработчиков**Использовать стек технологий для работы: Python, Django, PostgreSQL, Rest API, Nginx, Яндекс Облако и при желании и необходимости Celery (необязательное требование). Подготовить структуру базы данных и архитектуру сервиса. Реализовать логику, на основе интерфейсов дизайнеров и технического задания. Интегрировать бэкенд-часть сервиса с командой frontend разработчиков.