**СОДЕРЖАНИЕ**

НАИМЕНОВАНИЕ 3

ЦЕЛЬ 3

ЗАДАЧИ 3

ГЛОСАРИЙ 3

ВВЕДЕНИЕ 3

АРХИТЕКТУРА 4

РАЗРАБОТКА 4

1. Первичная настройка 4
2. Создание назначений 5

ПРИМЕР РАБОТЫ 9

РЕЗУЛЬТАТЫ 9

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 10

# **НАИМЕНОВАНИЕ**

“Бот для корпоративного мессенджера «Slack»”.

# **ЦЕЛЬ**

Бот должен принимать от пользователя запрос, состоящий из следующих параметров: имя, вид транспорта, страна, дата, email и моб. телефон. При этом, если пользователь ранее упоминал один из этих параметров — помнить его в рамках контекста, а также сохранять историю переписки.

# **ЗАДАЧИ**

Написать Бот для «Slack» с определенными функциями и навыками.

# **ГЛОСАРИЙ**

*Dialogflow* — это сервис, позволяющий создавать чат-ботов для разных платформ и языков на разных устройствах.

*Entities* —это группа объектов, которые необходимо распознать агенту.

*Intents*— это сопоставление между тем, что говорит пользователь, и какие операции может выполнить бот.

*Required* − определяет, является ли этот параметр обязательным для действия.

*Entity*  − сопоставление параметров с сущностями.

*Value* − если мы хотим повторно использовать этот параметр в ответе, можно просто использовать имя его значения.

*Is List*  − может ли значение параметра быть списком (например, в заказе может быть более одного параметра).

*Prompts* − клише.

# **ВВЕДЕНИЕ**

Создание бота которой выполняет функции тур-агента собирающий основные данные (на английском языке) пользователя, которые могут быть использованы любым человеком использующим этот бот в качестве дальнейшего оформления путевки.

# **АРХИТЕКТУРА**

Устройство чат-бота представлена на картинке ниже:

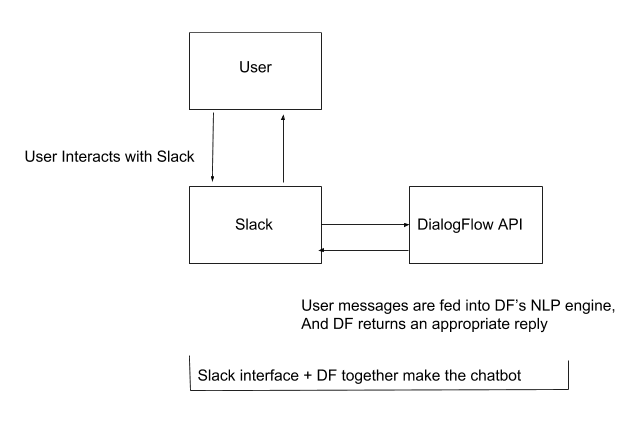


Рисунок 1 – Устройство чат-бота.

Используем API Slack для отправки сообщений в DialogFlow (DF). NLP-движок будет принимать информацию о намерении совершить действие и распознавать смысл сообщения в целом, а затем формировать релевантный ответ для пользователя. Интерфейс Slack и DialogFlow вместе формируют чат-бота.

# **РАЗРАБОТКА**

1)Первичная настройка:

Для начала регистрируемся в DialogFlow используя свой аккаунт Google и создаем нового агента:

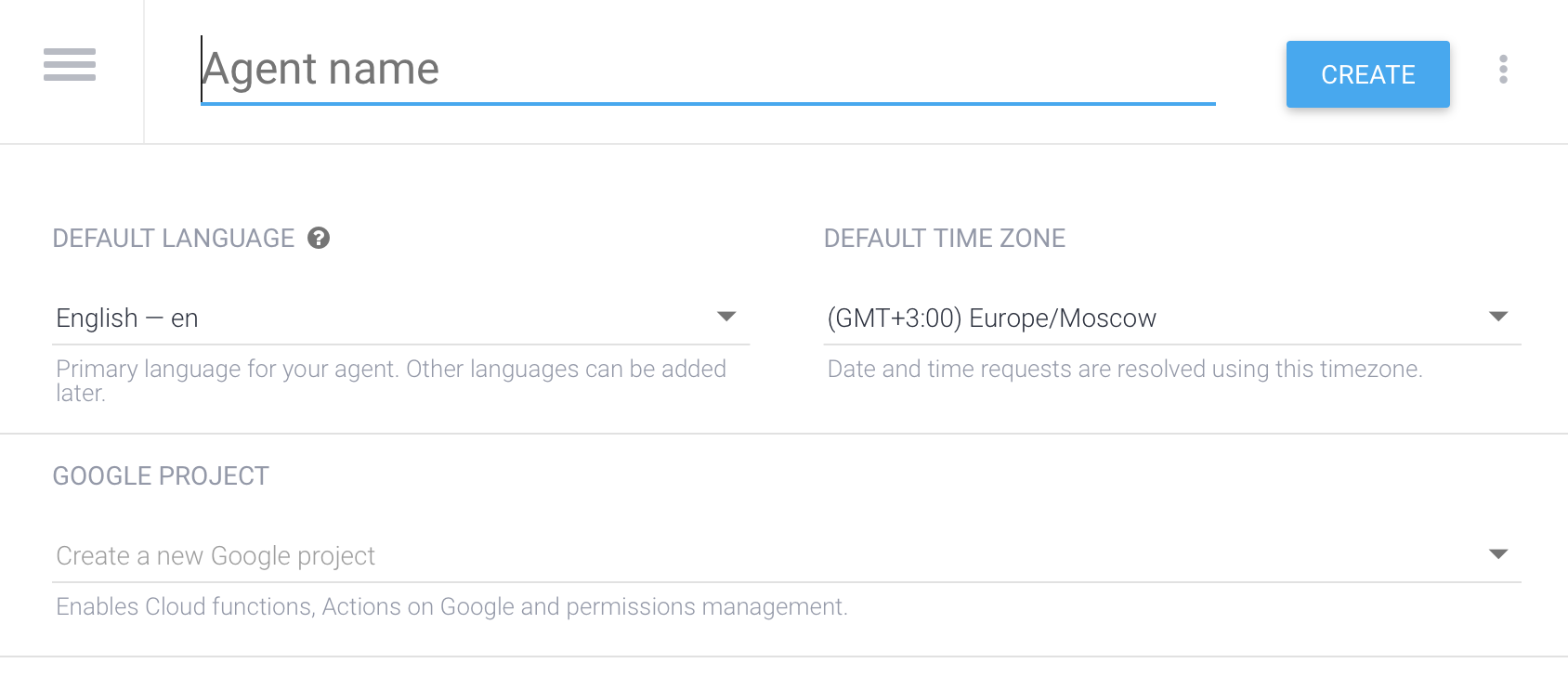


Рисунок 2 – Регистрация нового агента.

-Название агенты выбираем любое на ваш вкус.

-Выбираем язык (в нашем случае «Slack» распознает только англ. яз.)

-Временная зона очень важна, т.к. мы в дальнейшем можем давать указания боту используя такие термины как «Завтра» и подобные.

После создания агента мы переходим на вкладку «Integrations» и включаем все нужные нам платформы (в нашем случае это «Slack»), на которых наш бот будет успешно работать.

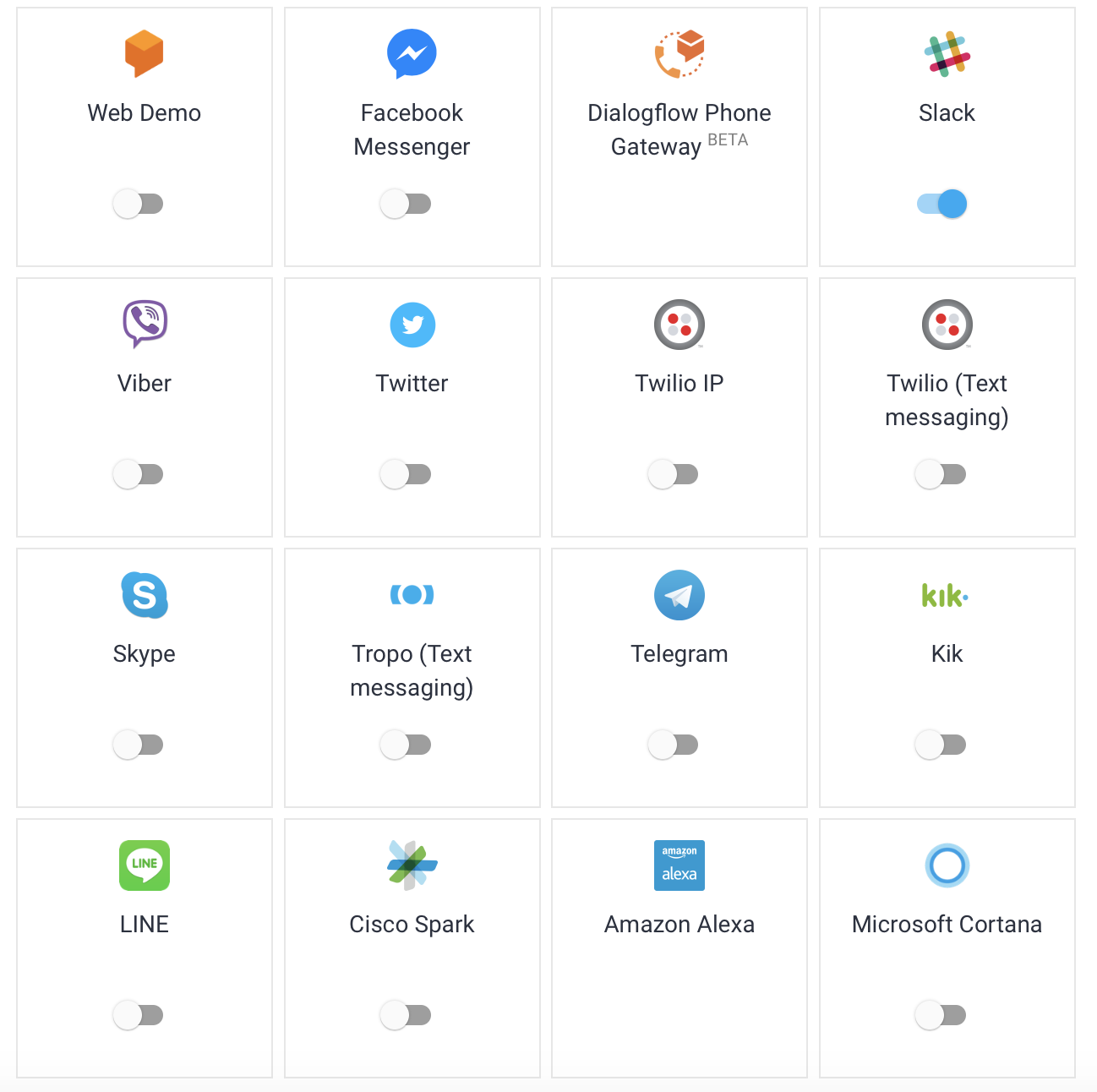


Рисунок 3 – Список интеграций которыми мы можем воспользоваться.

На данном этапе боту уже запущен и может общаться с пользователем в форме «привет-пока». Теперь нам нужно обучить его нужным и правильном пониманием фраз.

2)Создание назначений:

Назначение — это обработчик определённого запроса от пользователя.

Переходим на вкладку «Intents». После создания агента здесь будут два стандартных назначения:

Default Welcome Intent — запускается для приветствия пользователя (отправит ответное приветствие).

Default Fallback Intent — запускается в том случае, если ни одно из других назначений не подходит. Отправит сообщение, что пользователя не поняли.

Открываем «Default Welcome Intent». Здесь есть несколько секций: Contexts, User says, Events, Action, Response. Обратите внимание на секцию «Events»: здесь указано одно служебное событие «WELCOME» (назначение можно запустить двумя способами: как реакцию на текст пользователя или по событию).

После мы можем добавить приветственные фразы в «Training phrases», а затем в «Responses» вписываем соответствующие ответные фразы бота которыми он будет отвечать в случайном порядке. После всего мы обязательно не забываем нажать на кнопку «Save».

Теперь, мы возвращаемся обратно и открываем уже «Default Fallback Intent». Здесь мы прописываем фразы которые будут отображаться пользователю в случае не понимания.

Создадим назначения, которые будут реагировать на сообщения пользователя, в последующем выделяя каждый интересующий нас фрагмент различными цветными маркерами. Заходим в «Entities» и создаем новую сущность которую агент будет распознавать прямо внутри текста.

Фрагмент представленный ниже показывает последнюю стадию этого пункта:

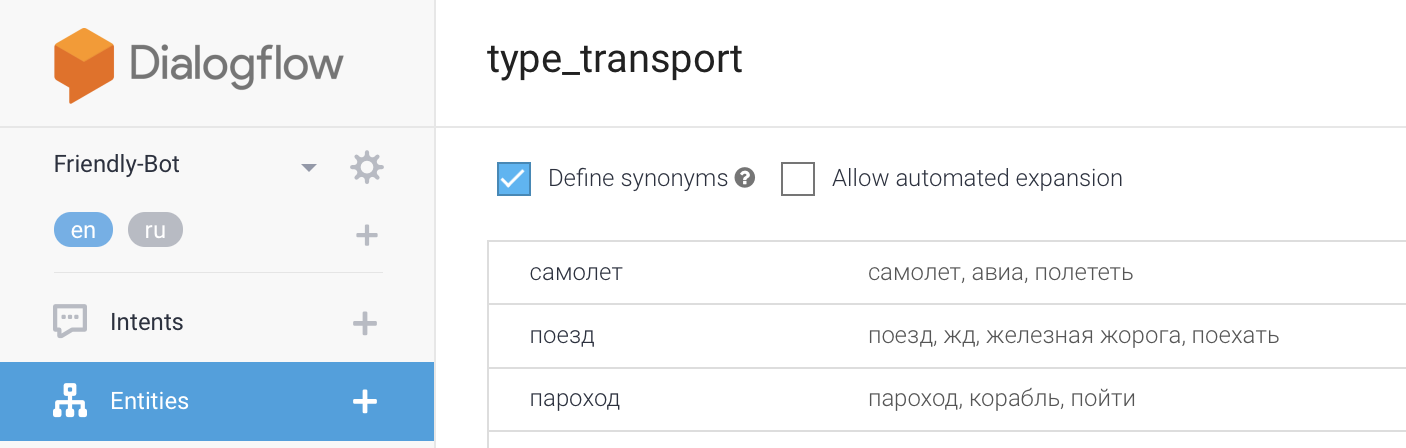


Рисунок 4 – Созданная сущность.

После мы снова возвращаемся в «Intents» нажимаем на кнопку «Создать новое намерение» и называем его «Заказ». Пролистываем вниз до «Action and parameters» прописываем все нужные нам параметры после чего указываем систему или сущность разработчика, каждый из которых будет соответствовать своему параметру. В «Prompts» мы пишем наводящие вопросы, которые соберут нужную информацию от пользователя.

На следующем изображении будет показано конечный результат на данном моменте:

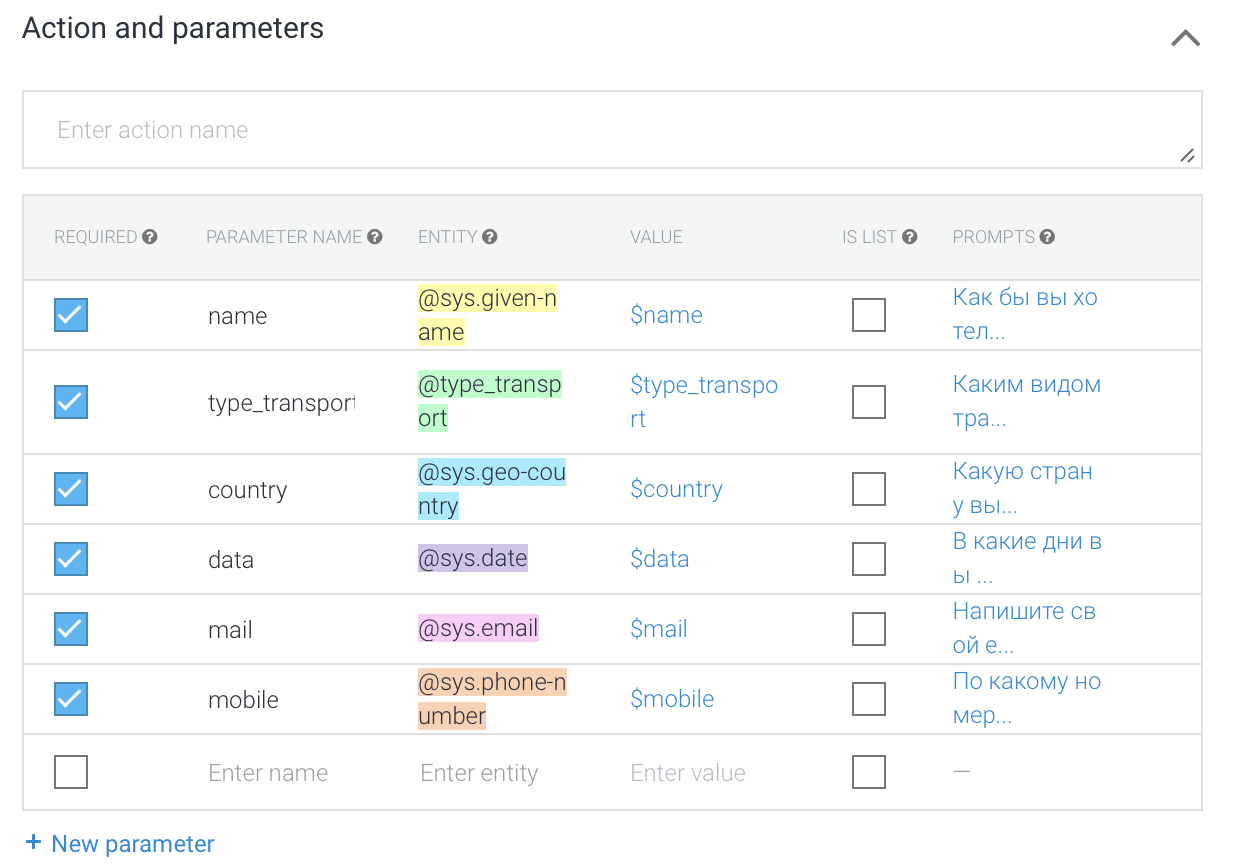


Рисунок 5 – Заполненная таблица «Action and parameters».

Далее мы прописываем те фразы, которые часто встречаются в нашей разговорной речи в разделе «Training phrases». Способ работы DialogFlow заключается в том, что вы можете использовать некоторые примеры пользовательских запросов и обучить систему реагировать на них по-разному в зависимости от контекста. Идея состоит в том, чтобы подготовить для агента как можно более разнообразные примеры, тогда и ответы будут более качественными. После настройки обучающих фраз, вам нужно определить, какое действие предпримет агент, при получении этого намерения. Все эти действия будут помечены тем же цветом которым ранее были заданы.

Итого получаем что-то подобное:

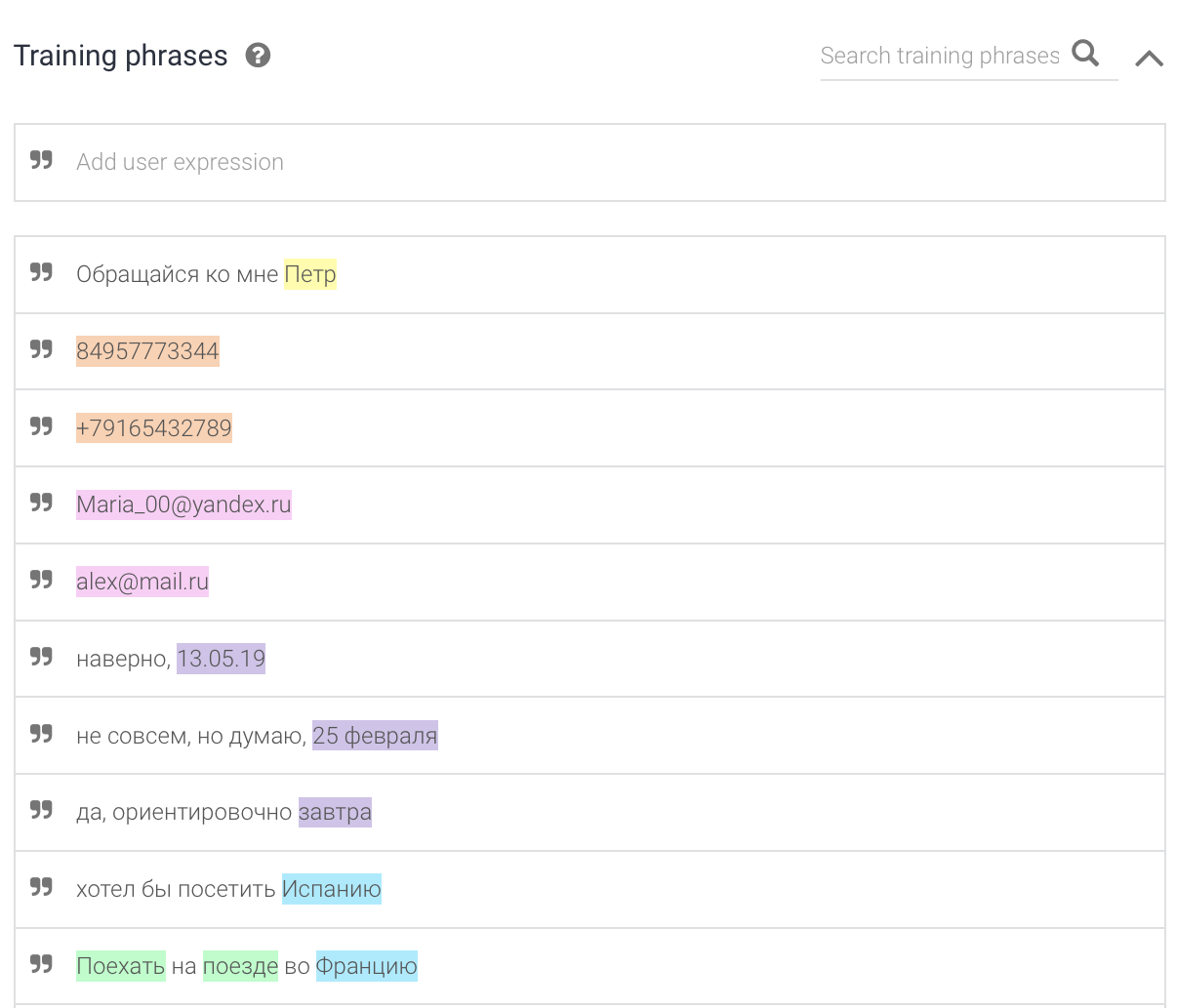


Рисунок 6 – Фрагмент обучающих фраз.

И чтобы после всех собранных данных пользователь получил ответ, нужно спуститься в самый низ страницы, где мы увидим раздел «Responses». Там мы прописываем фразу, которая будет завершительной в нашем диалоге с пользователем, а так же можно заметить, мы можем использовать любой из параметров которые автоматически заменятся значениями параметров, извлеченными из беседы с пользователем.

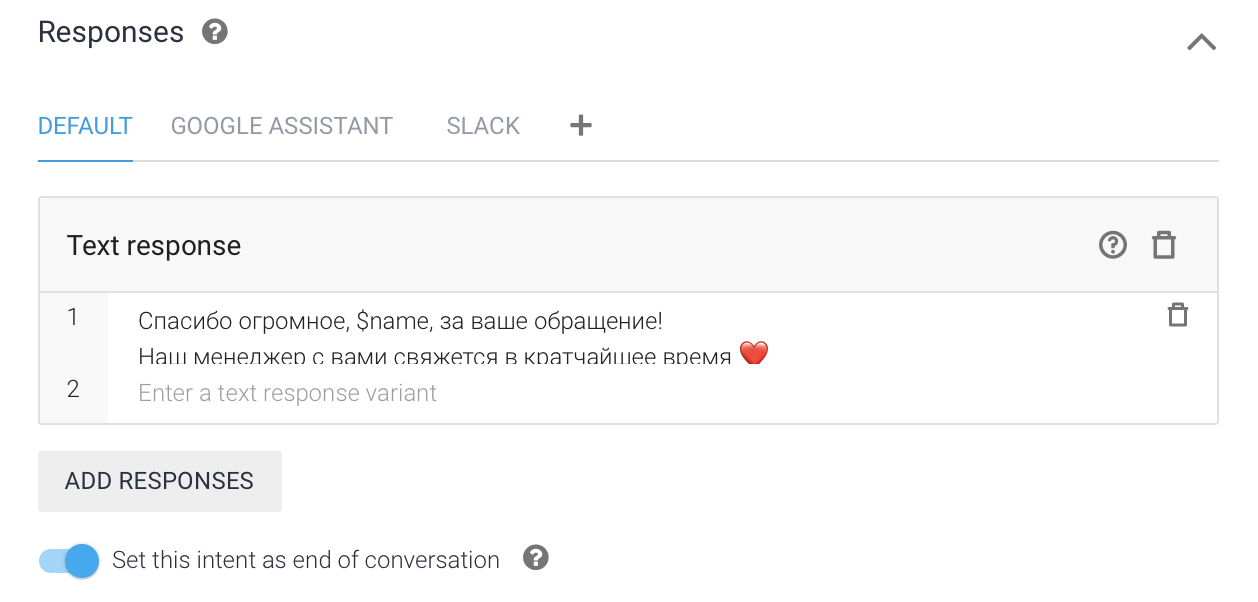


Рисунок 7 – Завершительная фраза нашего диалога

# **ПРИМЕР РАБОТЫ**

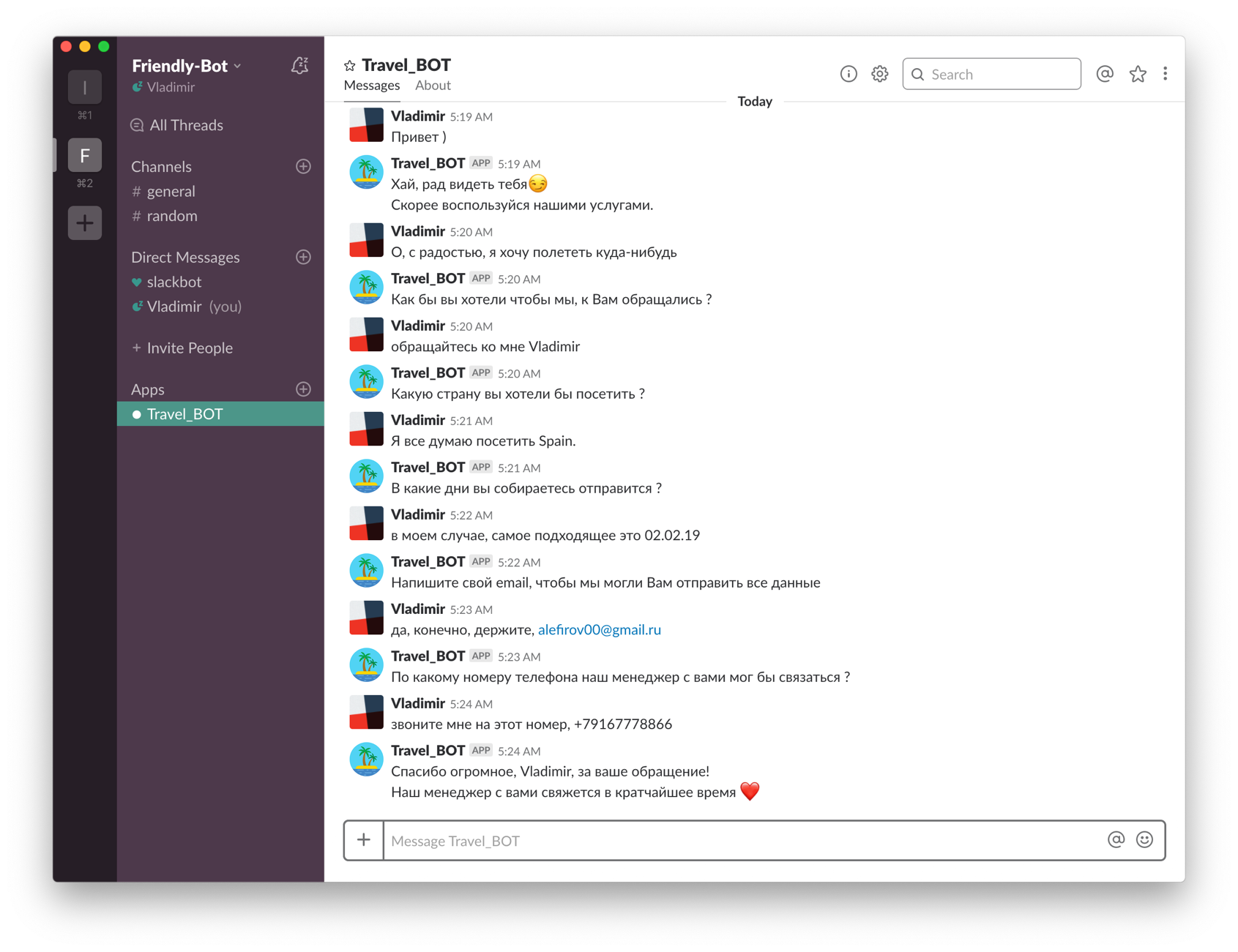


Рисунок 8 – Фрагмент диалога с нашим ботом

# **РЕЗУЛЬТАТЫ**

Поставленная цель выполнена успешна, со всеми задачами справился, в будущем цели подключить сторонний сервер через API, а также подключить базу данных, которая будет собирать в отдельные строки и столбцы пользователей, которые воспользуются данным ботом.

# **СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ**

1. Ознакомление уже с имеющимися ботами в Slack [Электронный ресурс]. – Режим доступа: URL: <https://vc.ru/flood/14035-slack-bots>. (дата обращения: 11.09.2018)
2. Чат-бот понимающий человеческую речь на Dialogflow [Электронный ресурс]. – Режим доступа: URL: <https://habr.com/post/342762/>. (дата обращения: 15.09.2018)
3. Создаём чат-ботов для Slack: от простых до обученных ИИ [Электронный ресурс]. – Режим доступа: URL: <https://proglib.io/p/intelligent-slack-bot/>. (дата обращения: 22.09.2018)
4. Диалоговый поток поддерживаемый Google [Электронный ресурс]. – Режим доступа: URL: <https://dialogflow.com>. (дата обращения: 06.10.2018)
5. Slack API [Электронный ресурс]. – Режим доступа: URL: <https://api.slack.com>. (дата обращения: 13.10.2018)
6. Требование к курсовому проекту [Электронный ресурс]. – Режим доступа:URL:<https://gist.github.com/drewxa/61053007c05d4e03811b919d396632ce>. (дата обращения: 20.10.2018)
7. Форум русскоязычный Dialogflow [Электронный ресурс]. – Режим доступа: URL: <https://dialogflow.ru/index.php>. (дата обращения: 27.10.2018)
8. Сохранение истории диалогового окна [Электронный ресурс]. – Режим доступа: URL: <http://qaru.site/questions/15452181/how-to-store-and-retrieve-the-chat-history-of-the-dialogflow> (дата обращения: 10.11.2018)