

Техническое задание на разработку телеграмм-бота для демонстрации возможностей облачного решения

Цель: Разработать демонстрационное приложение.

Необходимо реализовать Телеграмм-бота на языке C# с использованием

`FastReport.Cloud.SDK` для демонстрации возможностей облачного решения `FastReport Cloud`.

Основные требования

Требование к языку программирования и фреймворку: C#, .NET 5.

Требование к Боту:

1. Бот должен принимать 3 команды:
 - о Команда `/help` должна возвращать короткое описание, список доступных команд, информацию об авторах приложения и ссылку на <https://fastreport.cloud>.
 - о Команда `/start` должна возвращать короткое описание.
 - о Команда `/status` должна возвращать информацию о текущем статусе выполнения задачи.
3. Бот должен быть реализован с использованием диалога. Помимо команд в чате, должен быть реализован механизм Telegram Buttons <https://core.telegram.org/api/bots/buttons#callback-queries>. Для каждой команды следует реализовать кнопку.
4. На вход Бот должен принимать FRX или FPX файлы, после загрузки Бот должен сообщить пользователю, что файл добавлен в очередь и выдать кнопку запроса статуса.
5. На выходе Бот должен отправить в чат сформированный PDF файл.
6. Если в процессе обработки данных произошла ошибка, Бот должен информировать пользователя.

Дизайн и текст:

1. Орфографические ошибки не допускаются.
2. Интерфейс Бота должен быть лаконичным и удобным.
3. Дизайн и текст остаются на усмотрение разработчиков.

Примечания:

1. Необходимо предусмотреть случай, когда пользователь загружает отличный от FRX или FPX файл. В этом случае необходимо отправить пользователю информацию об ошибке.
2. Предусмотреть очередь для каждого пользователя. Если отчет уже отправлен на построение в PDF, то следующий файл должен ожидать в очереди.
3. Для отладки можно использовать файл: <https://raw.githubusercontent.com/FastReports/FastReport/master/Demos/Reports/Text.frx>
4. Для отладки и демонстрации рекомендуем использовать обратный туннель <https://ngrok.com/>.
5. Для доступа к API можно использовать nuget пакет FastReport.Cloud.SDK.
6. Документация доступна по адресу https://fastreport.cloud/ru/docs/guides/c_sharp.
7. Демонстрационные примеры <https://github.com/FastReports/FastReport.Cloud/Demos>.

8. Токены доступа к FastReport Cloud необходимо получить у экспертов компании Fast Reports.

Дополнительное задание № 1

Разработчик может использовать любой другой язык программирования без использования FastReport.Cloud.SDK, в этом случае остальные требования сохраняются. Документация: https://fastreport.cloud/ru/docs/guides/rest_api.

Дополнительное задание № 2

Реализовать возможность использования параметров отчёта. Диалог взаимодействия и способы ввода параметров остаются на усмотрение разработчика.

Документация: https://www.fast-report.com/public_download/docs/FRNet/online/en/UserManual/en-US/Data/ReportParameters.html

Дополнительное задание № 3

Реализовать возможность получения результата не только в PDF, но и в других форматах: RTF, DOCX, XLSX, SVG и пр.

Требования к приватности

`ApiKey` или любой другой токен доступа к `FastReport Cloud` не должен быть опубликован.

Полезные ссылки

1. Документация FastReport Cloud <https://fastreport.cloud/ru/Docs/guides>
2. FastReport Open Source (FastReport Designer Community - в релизах) <https://github.com/FastReports/FastReport/>
3. Документация FastReport .NET https://www.fastreport.ru/public_download/docs/FRNet/online/ru/UserManual/ru-RU/Fundamentals/TheReport/TheDesigner.html
4. FastReport .NET <https://www.fast-report.com/en/product/fast-report-net/>
5. Ngrok <https://ngrok.com/>
6. Telegram Buttons <https://core.telegram.org/api/bots/buttons#callback-queries>

Контакт с экспертами Fast Reports

Группа в Телеграмм <https://t.me/joinchat/iyANiX83b-o5NWRi>