

# File Parser

## Objective

The objective is to create file parser for txt files. The files are structured.

The performance of the solution will be tested on specific file structures that will be given to the developer.

The file parser shall be developed in several implementation steps. This document outlines the step 1, where it is OK to create a tokenless /scannerless parser for specific form of txt file.

The final parser solution will be architected in a way to support multiple input formats, besides txt also XML, JSON. This implies that we are looking for lexer-parser architecture.

## Used technologies

The technologies to be used are:

- C# as programming language
- .NET Core or .NET Framework 2.1
- preferably MS SQL as SQL
- Linux Ubuntu as OS, details to be agreed

In agreement with us it is allowed to use other technologies when they come in executable packages that are not modified.

## Requirements

### Constraints

It is preferred to use one of the existing C# parsers (parser libraries) as starting point. The exceptions from this requirements must be approved by the contractor (us).

In the first implementation step we implement parser for one specific txt file. The file will be given and the information that we want to extract will be clearly defined.

Before implementation start, we require that developer consults with us about the architecture.

### Method and use cases

The goal is to create method(s) that will cover the following use case:

1. Give the method a tag and the method will find the corresponding value in file and return it as a JSON object example: "temperature": "35°C"
2. Give the method list of tags and the method will find the corresponding values in file and return them as list of JSON object
3. Give the method two tags (two delimiters) and the method will return the content that it found between the two delimiters as text string including line breaks

Therefore we see the method(s) have the following inputs:

- absolute file path to local file system (of the file to be parsed)
- file name (of the file to be parsed)
- optionally file classification label (to know which file schema/structure one shall apply) - if you create a method that needs this
- the tag(s)

The method shall return:

- the result object = JSON object
- the result with result code

The applicable result codes are listed below

| Result Code | Result meaning                                  |
|-------------|---|
| 0           | Success, tag found, result object returned      |
| 1           | Error, tag not found                            |
| 2           | Error, tag found, but result object not created |
| 3           | Error, internal process error                   |

## Value tags

The values tags are indicated in the attached files.

[PANELBOX.pdf](#)