SafeCape Quisper API documentation

VERSION 2 – DECEMBER 2015

Contents

Contact Information

Provider: SafeCape Software Solutions
Email: RulesToolset@SafeCape.gr
Web site: www.safecape.gr

Introduction

This documentation contains general information about SafeCape's REST API available through the Quisper© platform and examples (sample code) in C#.

SafeCape API is powered by Rules Toolset© technology and can be used to make API calls targeting different products. Each **product** is a domain-specific model comprising of a set of rules that can be used to evaluate a set of **input variables** and produce a set of **output variables**. The product named "Qualify", which contains scientifically validated genotype-phenotype-nutrition associations, is available (/safecape/products/qualify/execute) and can be used to generate personalized nutrition and lifestyle recommendations for an individual.

The user must pass a valid **user_key** custom header to the service or else the request is not fulfilled and a *403 Forbidden* HTTP status code is returned. All successful requests return *200 OK* HTTP status code.

Before running the C# examples, you have to install the Microsoft ASP.NET Web API 2.2 Client package by issuing the following command in the Package Manager Console:

```
PM> Install-Package Microsoft.AspNet.WebApi.Client
```

Base URLS

All URLs referenced in the documentation have the following base:

http://api.quisper.eu

1. Get the list of all available products

Resource URI

```
/safecape/products
```

HTTP GET

```
curl -G "http://api.quisper.eu/safecape/products" -H "user_key: YOUR_USER_KEY"

{
```

```
  "products": [
    "EuroFIT_DNAFit_ENGLISH",
    "Qualify",
    "Nutrigene_DIET_2_ADRB27"
  ]
}
```

C# Example

```csharp
using System.Text;
using System.Threading.Tasks;
using Newtonsoft.Json;
using Newtonsoft.Json.Linq;
using System.Xml.Serialization;
using System.Xml.Linq;
using System.Xml;
using System.IO;

namespace QualifyAPIClient
{
    public class Program
    {
        static void Main()
        {
            RunAsync();
            Console.ReadLine();
        }

        static async void RunAsync()
        {
            using (var client = new HttpClient())
            {
                client.BaseAddress = new Uri("http://api.quisper.eu/");
                client.DefaultRequestHeaders.Add("user_key", "YOUR_USER_KEY");
                string requestUri = "safecape/products";
                HttpResponseMessage response = await client.GetAsync(requestUri);
                if (response.IsSuccessStatusCode)
                {
                    JToken results = await response.Content.ReadAsAsync<JToken>();
                    JArray products = (JArray)results["products"];
                    foreach (var item in products)
                        Console.WriteLine(item.ToString());
                }
                else
                {
                    Console.WriteLine(string.Format("HTTP Status Code: {0} {1}",
response.StatusCode, response.Content.ReadAsStringAsync().Result));
                }
            }
        }
    }
}
```

2. Get the available versions of a product and their status

Resource URI

/safecape/products/{productName}

HTTP GET

curl -G "http://api.quisper.eu/safecape/products/qualify" -H "user_key: YOUR_USER_KEY"

{

pg. 3

```
  "Name": "Qualify",
  "Versions": [
    {
      "Version": 1,
      "Status": "draft"
    }
  ]
}
```

C# Example

```csharp
using System;
using System.Collections.Generic;
using System.Net.Http;
using System.Text;
using System.Threading.Tasks;
using Newtonsoft.Json;
using Newtonsoft.Json.Linq;
using System.Xml.Serialization;
using System.Xml.Linq;
using System.Xml;
using System.IO;

namespace QualifyAPIClient
{
    public class Program
    {
        static void Main()
        {
            RunAsync();
            Console.ReadLine();
        }

        static async void RunAsync()
        {
            using (var client = new HttpClient())
            {
                client.BaseAddress = new Uri("http://api.quisper.eu/");
                client.DefaultRequestHeaders.Add("user_key", "YOUR_USER_KEY");

                string requestUri = "safecape/products/qualify";
                HttpResponseMessage response = await client.GetAsync(requestUri);
                if (response.IsSuccessStatusCode)
                {
                    JToken results = await response.Content.ReadAsAsync<JToken>();
                    JArray versions = (JArray)results["Versions"];
                    foreach (var item in versions)
                        Console.WriteLine(item.ToString());
                    JObject oVersion = (JObject)versions[0];
                    Console.WriteLine(string.Format("Current version is '{0}' and status is
'{1}'.",
                        oVersion.SelectToken("Version"), oVersion.SelectToken("Status")));
                }
                else
                {
                    Console.WriteLine(string.Format("HTTP Status Code: {0} {1}",
response.StatusCode, response.Content.ReadAsStringAsync().Result));
                }
            }
        }
    }
}
```

3. Apply the rules of a product (model) to the JSON-formatted input values

Resource URI

/safecape/products/{productName}/execute

HTTP POST for this resource evaluates the JSON-formatted input values compared to the model (product) specified by the parameter *{productName}* e.g. "qualify" and produces a JSON-formatted output according to the rules of the product. Both input and output represent a case, which is organized in concepts that contain variables (name-value pairs).

HTTP POST

```
curl -XPOST "http://api.quisper.eu/safecape/products/qualify/execute"
-H "Content-Type: application/json; charset=utf-8"
-H "user_key: YOUR_USER_KEY"
-d @input.json > output.json
```

Example Input (input.json)

The JSON-encoded input and output data consist of name-value pairs (variables) organized in concepts. Input includes the concept "Personal_Details" with phenotype information, such as gender, weight, height etc. and the concept "Genes" containing the individual's genotype.

```
{
  "Name": "Qualify",
  "Use": null,
  "Concepts": [
    {
      "Name": "Personal_Details",
      "Variables": [
        {
          "Name": "Gender",
          "Value": "M"
        },
        {
          "Name": "Name",
          "Value": "SAMPLE101"
        },
        {
          "Name": "DoB",
          "Value": "08/07/1975"
        },
        {
          "Name": "Weight",
          "Value": "81"
        },
        {
          "Name": "Height",
          "Value": "1.83"
        },
        {
```

pg. 5

```
        "Name": "Age",
        "Value": "54"
      },
      {
        "Name": "Activity_level",
        "Value": "low"
      }
    ],
    "Concepts": [

    ]
  },
  {
    "Name": "Genes",
    "Variables": [
      {
        "Name": "APOC3_result",
        "Value": "GG"
      },
      {
        "Name": "ADH1C_result",
        "Value": "AG"
      },
      {
        "Name": "ADRB2_AG_ArgGly_result",
        "Value": "GG"
      },
      {
        "Name": "ADRB2_CG_GlnGlu_gene_result",
        "Value": "CG"
      },
      {
        "Name": "ADRB3_CT_ArgTrp_result",
        "Value": "CT"
      },
      {
        "Name": "APOA2_265TC_result",
        "Value": "CC"
      },
      {
        "Name": "ACE_CG_ID_result",
        "Value": "CG"
      },
      {
        "Name": "AGTmet_thr_result",
        "Value": "TT"
      },
      {
        "Name": "ACTN3_CT_result",
        "Value": "TT"
      },
      {
        "Name": "BDKRB2_99_result",
        "Value": "CC"
      },
      {
        "Name": "COL1A1_GT_result",
        "Value": "GG"
      },
      {
        "Name": "COL5A1_CT_result",
        "Value": "TT"
      },
      {
        "Name": "CRP_1082GA_result",
        "Value": "CC"
      },
      {
        "Name": "CYP1A2_result",
        "Value": "AA"
      },
      {
        "Name": "CAT_C262T_result",
        "Value": "AG"
```

```
    },
    {
      "Name": "CETP_G279A_result",
      "Value": "ND"
    },
    {
      "Name": "EPHX1_Tyr113His_result",
      "Value": "CT"
    },
    {
      "Name": "FABP2_GA_AlaThr_result",
      "Value": "CC"
    },
    {
      "Name": "FTO_TA_result",
      "Value": "TT"
    },
    {
      "Name": "GDF5_CT_result",
      "Value": "CT"
    },
    {
      "Name": "GPX_Pro198Leu_result",
      "Value": "CC"
    },
    {
      "Name": "GSTM1_result",
      "Value": "D"
    },
    {
      "Name": "GSTT1_result",
      "Value": "I"
    },
    {
      "Name": "IL6_result",
      "Value": "CG"
    },
    {
      "Name": "IL6R_AC_result",
      "Value": "AA"
    },
    {
      "Name": "LPL_result",
      "Value": "CC"
    },
    {
      "Name": "LCT_result",
      "Value": "CT"
    },
    {
      "Name": "MTHFR_result",
      "Value": "CT"
    },
    {
      "Name": "NRF_AG_result",
      "Value": "AA"
    },
    {
      "Name": "NOS3_G894T_result",
      "Value": "ND"
    },
    {
      "Name": "PPARG_CG_result",
      "Value": "CC"
    },
    {
      "Name": "PPARA_GC_result",
      "Value": "CG"
    },
    {
      "Name": "PPARGC1A_GA_result",
      "Value": "AG"
    },
    {
```

```json
      "Name": "TCF7L2_CT_result",
      "Value": "TT"
    },
    {
      "Name": "TNF_result",
      "Value": "AG"
    },
    {
      "Name": "TRHR_TG_result",
      "Value": "AA"
    },
    {
      "Name": "SOD2_result",
      "Value": "TT"
    },
    {
      "Name": "VDR_result",
      "Value": "TT"
    },
    {
      "Name": "VEGF_CG_result",
      "Value": "CG"
    },
    {
      "Name": "rs2395182_DQA1201_result",
      "Value": "TT"
    },
    {
      "Name": "rs7775228_DQB1202_result",
      "Value": "TT"
    },
    {
      "Name": "rs4713586_DQ4_result",
      "Value": "TT"
    },
    {
      "Name": "rs2187668_DQ25_result",
      "Value": "GG"
    },
    {
      "Name": "rs4639334_DQ7_result",
      "Value": "AA"
    },
    {
      "Name": "rs7454108_DQ8_result",
      "Value": "TT"
    }
  ],
  "Concepts": [

  ]
    }
  ]
}
```

Example Output (output.json)

```json
{
  "ModelOutput": {
    "Name": "Qualify",
    "Use": "Output",
    "Concepts": [
      {
        "Name": "Personal_Details",
        "Variables": [
          {
            "Name": "Gender",
            "Value": "M"
          },
          {
            "Name": "Name",
            "Value": "SAMPLE101"
          },
          {
            "Name": "DoB",
            "Value": "08/07/1975"
          },
          {
            "Name": "Weight",
            "Value": "81"
          },
          {
            "Name": "Height",
            "Value": "1.83"
          },
          {
            "Name": "Project_number",
            "Value": ""
          },
          {
            "Name": "BMIround",
            "Value": "24.2"
          }
        ],
        "Concepts": []
      },
      {
        "Name": "Genes",
        "Variables": [
          {
            "Name": "ACE_CG_ID",
            "Value": "ID"
          },
          {
            "Name": "ACE",
            "Value": "ID"
          },
          {
            "Name": "APOC3",
            "Value": "GG"
          },
          {
            "Name": "ADH1C",
            "Value": "AG"
          },
          {
            "Name": "ADRB2_AG_ArgGly",
            "Value": "GG"
          },
          {
            "Name": "ADRB2_CG_GlnGlu_gene",
            "Value": "CG"
          },
          {
            "Name": "ADRB3_CT_ArgTrp",
            "Value": "CT"
```

```json
    },
    {
      "Name": "APOA2_265TC",
      "Value": "CC"
    },
    {
      "Name": "AGTmet_thr",
      "Value": "TT"
    },
    {
      "Name": "ACTN3_CT",
      "Value": "TT"
    },
    {
      "Name": "BDKRB2_99",
      "Value": "CC"
    },
    {
      "Name": "COL1A1_GT",
      "Value": "GG"
    },
    {
      "Name": "COL5A1_CT",
      "Value": "TT"
    },
    {
      "Name": "CRP_1082GA",
      "Value": "GG"
    },
    {
      "Name": "CYP1A2",
      "Value": "AA"
    },
    {
      "Name": "CAT_C262T",
      "Value": "TC"
    },
    {
      "Name": "CETP_G279A",
      "Value": "ND"
    },
    {
      "Name": "EPHX1_Tyr113His",
      "Value": "CT"
    },
    {
      "Name": "FABP2_GA_AlaThr",
      "Value": "GG"
    },
    {
      "Name": "FTO_TA",
      "Value": "TT"
    },
    {
      "Name": "GDF5_CT",
      "Value": "CT"
    },
    {
      "Name": "GPX_Pro198Leu",
      "Value": "CC"
    },
    {
      "Name": "GSTM1",
      "Value": "D"
    },
    {
      "Name": "GSTT1",
      "Value": "I"
    },
    {
      "Name": "IL6",
      "Value": "CG"
    },
    {
```

```json
    "Name": "IL6R_AC",
    "Value": "AA"
},
{
  "Name": "LPL",
  "Value": "CC"
},
{
  "Name": "LCT",
  "Value": "CT"
},
{
  "Name": "MTHFR",
  "Value": "CT"
},
{
  "Name": "NRF_AG",
  "Value": "AA"
},
{
  "Name": "PPARG_CG",
  "Value": "CC"
},
{
  "Name": "PPARG",
  "Value": "CC"
},
{
  "Name": "PPARG_CG_ProAla",
  "Value": "CC"
},
{
  "Name": "PPARA_GC",
  "Value": "CG"
},
{
  "Name": "PPARGC1A_GA",
  "Value": "AG"
},
{
  "Name": "TCF7L2_CT",
  "Value": "TT"
},
{
  "Name": "TNF",
  "Value": "AG"
},
{
  "Name": "TRHR_TG",
  "Value": "TT"
},
{
  "Name": "SOD2",
  "Value": "TT"
},
{
  "Name": "VDR",
  "Value": "TT"
},
{
  "Name": "VEGF_CG",
  "Value": "CG"
},
{
  "Name": "rs2395182_DQA1201",
  "Value": "TT"
},
{
  "Name": "rs7775228_DQB1202",
  "Value": "TT"
},
{
  "Name": "rs4713586_DQ4",
  "Value": "TT"
```

```
        },
        {
          "Name": "rs2187668_DQ25",
          "Value": "GG"
        },
        {
          "Name": "rs4639334_DQ7",
          "Value": "AA"
        },
        {
          "Name": "rs7454108_DQ8",
          "Value": "TT"
        },
        {
          "Name": "DQ22",
          "Value": ""
        },
        {
          "Name": "DQ25",
          "Value": ""
        },
        {
          "Name": "DQ7",
          "Value": "DQ7/DQ7"
        },
        {
          "Name": "DQ8",
          "Value": ""
        },
        {
          "Name": "DQA1_201allele",
          "Value": "DQA1-201/DQA1-201"
        },
        {
          "Name": "DQB1_202allele",
          "Value": "0"
        },
        {
          "Name": "DQ25allele",
          "Value": ""
        },
        {
          "Name": "DQ7allele",
          "Value": "DQA1-505, DQB1-301"
        },
        {
          "Name": "DQ8allele",
          "Value": ""
        },
        {
          "Name": "EPHX1_tyrhis",
          "Value": "Tyr/His"
        },
        {
          "Name": "GPX_ProLeu",
          "Value": "Pro/Pro"
        },
        {
          "Name": "VDR_genotype",
          "Value": "TT"
        },
        {
          "Name": "ACE_genotype",
          "Value": "ID"
        },
        {
          "Name": "AGT_genotype",
          "Value": "Met/Met"
        },
        {
          "Name": "ADHC1gene",
          "Value": "AG - Ile / Val"
        },
        {
```

```json
          "Name": "HLA_type",
          "Value": "   DQ7/DQ7 "
        },
        {
          "Name": "DQ_Status",
          "Value": "Negative"
        },
        {
          "Name": "ADRB2_Arg16Gly",
          "Value": "Gly-Gly"
        },
        {
          "Name": "ADRB2_Gln27Glu",
          "Value": "Gln-Glu"
        },
        {
          "Name": "ADRB3_Arg64Trp",
          "Value": "Arg-Trp"
        },
        {
          "Name": "FABP2_Ala54Thr",
          "Value": "Ala-Ala"
        },
        {
          "Name": "PPARG_ProAla",
          "Value": "Pro-Pro"
        },
        {
          "Name": "ADRB2_CG_GlnGlu",
          "Value": "CG"
        },
        {
          "Name": "ACE_AGT_Hap",
          "Value": "IDMet/Met"
        }
      ],
      "Concepts": [
        {
          "Name": "Gene_Impact",
          "Variables": [
            {
              "Name": "GSTM1_Impact",
              "Value": "**"
            },
            {
              "Name": "GSTT1_Impact",
              "Value": ""
            },
            {
              "Name": "TNF_Impact",
              "Value": "*"
            },
            {
              "Name": "IL6_Impact",
              "Value": "*"
            },
            {
              "Name": "MTHFR_Impact",
              "Value": "*"
            },
            {
              "Name": "PPARG_Impact",
              "Value": "**"
            },
            {
              "Name": "LPL_Impact",
              "Value": "*"
            },
            {
              "Name": "AGTsalt_Impact",
              "Value": ""
            },
            {
              "Name": "ACEsalt_Impact",
```

```json
        "Value": "*"
    },
    {
      "Name": "ACEcarb_Impact",
      "Value": "*"
    },
    {
      "Name": "SOD2_Impact",
      "Value": ""
    },
    {
      "Name": "APOC3_Impact",
      "Value": ""
    },
    {
      "Name": "VDR_Impact",
      "Value": ""
    },
    {
      "Name": "LCT_Impact",
      "Value": ""
    },
    {
      "Name": "ADH1C_Impact",
      "Value": ""
    },
    {
      "Name": "CYP1A2caffeine_Impact",
      "Value": ""
    },
    {
      "Name": "CYP1A2meat_Impact",
      "Value": "**"
    },
    {
      "Name": "TCF7L2_Impact",
      "Value": "**"
    },
    {
      "Name": "ADRB2GlnGlu_Impact",
      "Value": "*"
    },
    {
      "Name": "APOA2_impact",
      "Value": "**"
    },
    {
      "Name": "FTO_impact",
      "Value": ""
    },
    {
      "Name": "FABP2_impact_impact",
      "Value": ""
    },
    {
      "Name": "CAT_impact",
      "Value": "*"
    },
    {
      "Name": "EPHX1_impact",
      "Value": "*"
    },
    {
      "Name": "GPX1_impact",
      "Value": ""
    },
    {
      "Name": "ADRB2_FAT",
      "Value": "*"
    },
    {
      "Name": "ADRB3_FAT",
      "Value": "*"
    }
```

```
            ],
            "Concepts": []
        }
    ]
},
{
    "Name": "Overview",
    "Variables": [
        {
            "Name": "Increase_Folate",
            "Value": "Folic Acid, Vit B6 e B12"
        },
        {
            "Name": "Increase_Antiox",
            "Value": ""
        },
        {
            "Name": "Increase_VitD",
            "Value": ""
        },
        {
            "Name": "Increase_Calcium",
            "Value": ""
        },
        {
            "Name": "Increase_Cruciferous",
            "Value": "Cruciferous"
        },
        {
            "Name": "Increase_Omega3",
            "Value": "Omega 3"
        },
        {
            "Name": "Increase_Fibre",
            "Value": "Fiber"
        },
        {
            "Name": "Increase_Olive_Oil",
            "Value": ""
        },
        {
            "Name": "Decrease_Salt",
            "Value": "Salt"
        },
        {
            "Name": "Decrease_Caffeine",
            "Value": ""
        },
        {
            "Name": "Decrease_SatFat",
            "Value": "Saturated Fats"
        },
        {
            "Name": "Decrease_CHO",
            "Value": "Refined carbs / sugars"
        },
        {
            "Name": "Decrease_GrilledMeat",
            "Value": "Grilled Meat"
        },
        {
            "Name": "Lactose",
            "Value": "Lactose tolerant"
        },
        {
            "Name": "Celiac",
            "Value": "Negative"
        }
    ],
    "Concepts": []
},
{
    "Name": "Table_action",
    "Variables": [
```

pg. 15

```json
    {
      "Name": "salt",
      "Value": "Intermediate sensitivity to salt, <2,200 mg / day sodium"
    },
    {
      "Name": "Alcohol",
      "Value": "Positive effect of alcohol on cholesterol"
    },
    {
      "Name": "Olive_Oil",
      "Value": "Standard recommendations for olive oil"
    },
    {
      "Name": "Caffeine",
      "Value": "Standard recommendation"
    },
    {
      "Name": "Grilled_Meat",
      "Value": "Limit intake of grilled meat"
    },
    {
      "Name": "Cruciferous",
      "Value": "Consume 3-4 servings of cruciferous per week"
    },
    {
      "Name": "Inflammation",
      "Value": "Increased basal inflammation: 3 g Omega 3 / day"
    },
    {
      "Name": "Sat_Fat",
      "Value": "Limit saturated fat intake to < 16g / day"
    },
    {
      "Name": "Folate",
      "Value": "Intermediate: at least 400 µg folic acid, 10 mg Vit B6, 15 µg Vit B12
per day"
    },
    {
      "Name": "CHO",
      "Value": "Limit intake of refined carbohydrates: glycemic load <70 / day; consume
at least 25 g/day fibre"
    },
    {
      "Name": "Oxidative_stress",
      "Value": "Standard recommendation for antioxidants"
    },
    {
      "Name": "Selenium",
      "Value": ""
    },
    {
      "Name": "Nickel",
      "Value": "Increased predisposition for nickel sensitivity"
    },
    {
      "Name": "VitD",
      "Value": "Normal: 600 IU / day Vitamin D"
    },
    {
      "Name": "Lactose",
      "Value": "Lactose tolerant"
    },
    {
      "Name": "Lactose_a",
      "Value": "Normal for lactose"
    },
    {
      "Name": "Celiac",
      "Value": "Very low celiac disease risk"
    },
    {
      "Name": "Celiac_a",
      "Value": "Very low celiac disease risk"
    }
```

```
      ],
      "Concepts": []
    },
    {
      "Name": "Report_main",
      "Variables": [
        {
          "Name": "HLA_alleles",
          "Value": "DQ7/DQ7/]"
        },
        {
          "Name": "CHO_sensitivity",
          "Value": "high"
        },
        {
          "Name": "Lipid_sensitivity",
          "Value": "medium"
        }
      ],
      "Concepts": []
    },
    {
      "Name": "Goal_table",
      "Variables": [
        {
          "Name": "Vits_B",
          "Value": "*"
        },
        {
          "Name": "Antiox",
          "Value": ""
        },
        {
          "Name": "Selenium",
          "Value": ""
        },
        {
          "Name": "VitD",
          "Value": ""
        },
        {
          "Name": "Fibra",
          "Value": "*"
        },
        {
          "Name": "Omega3",
          "Value": ""
        },
        {
          "Name": "Calcium",
          "Value": ""
        },
        {
          "Name": "Salt",
          "Value": "*"
        },
        {
          "Name": "PhysicalActivity",
          "Value": "*"
        },
        {
          "Name": "Caffeine",
          "Value": ""
        },
        {
          "Name": "SatFat",
          "Value": "*"
        },
        {
          "Name": "CHO",
          "Value": "*"
        },
        {
          "Name": "Folate_goal",
```

```
          "Value": "400"
        },
        {
          "Name": "B6_goal",
          "Value": "8"
        },
        {
          "Name": "B12_goal",
          "Value": "10"
        },
        {
          "Name": "VitA_goal",
          "Value": "2,700 IU / 810 µg"
        },
        {
          "Name": "VitC_goal",
          "Value": "105"
        },
        {
          "Name": "VitE_goal",
          "Value": "15 IU / 13.5 mg"
        },
        {
          "Name": "VitD_goal",
          "Value": "600 IU / 5 µg"
        },
        {
          "Name": "Fibra_goal",
          "Value": "25"
        },
        {
          "Name": "Omega3_goal",
          "Value": "3"
        },
        {
          "Name": "Calcium_goal",
          "Value": "1000"
        },
        {
          "Name": "Salt_goal",
          "Value": "2.2"
        },
        {
          "Name": "Selenium_goal",
          "Value": "75"
        },
        {
          "Name": "PhysicalActivity_goal",
          "Value": "45 min  / day"
        },
        {
          "Name": "Caffeine_goal",
          "Value": "300"
        },
        {
          "Name": "SatFat_goal",
          "Value": "16"
        },
        {
          "Name": "CHO_goal",
          "Value": "70"
        }
      ],
      "Concepts": []
    },
    {
      "Name": "DietPanel",
      "Variables": [],
      "Concepts": [
        {
          "Name": "ExerciseEER",
          "Variables": [],
          "Concepts": [
            {
```

```
              "Name": "ExerciseCalories",
              "Variables": [
                {
                  "Name": "EERtotal",
                  "Value": "2680"
                },
                {
                  "Name": "BMIround",
                  "Value": "24.2"
                }
              ],
              "Concepts": []
            }
          ]
        },
        {
          "Name": "Scores",
          "Variables": [
            {
              "Name": "FatPercent_round",
              "Value": "5.3"
            },
            {
              "Name": "CarbPercent_round",
              "Value": "6.4"
            },
            {
              "Name": "ExercisePercent_round",
              "Value": "6.0"
            }
          ],
          "Concepts": []
        },
        {
          "Name": "Table_action",
          "Variables": [
            {
              "Name": "Refined_carbs_max",
              "Value": "6%"
            },
            {
              "Name": "Fibre_recc",
              "Value": "30"
            },
            {
              "Name": "SatFat_max",
              "Value": "6%"
            },
            {
              "Name": "Exercise_recc",
              "Value": "increased"
            },
            {
              "Name": "Exercise_recc_amount",
              "Value": "high"
            },
            {
              "Name": "Glicemic_load",
              "Value": "70"
            }
          ],
          "Concepts": []
        }
      ]
    }
  ]
}
}
```

pg. 19

Input Variables

CONCEPT		NAME	REQ	DATA TYPE	UNITS	POSSIBLE VALUES	DESCRIPTION	EXAM-PLE
Personal_Details	1	Gender	YES	string		M, F	The sex of the subject	
	2	Today_date		string			Current date	
	3	Name		string			The name of the subject	
	4	DoB		string			The birth date of the subject	
	5	Weight	YES	number	Kg		The weight of the subject	81
	6	Height	YES	number	m		The height of the subject	1.83
	7	Age	YES	number			The age of the subject	54
	8	Activity_level	YES	string		sedentary,low,medium, active	The activity level of the subject	
	9	Project_number		string				
Genes	1	APOC3_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs5128 (http://www.snpedia.com/index.php/rs5128)	
	2	ADH1C_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs698 (http://www.snpedia.com/index.php/rs698)	
	3	ADRB2_AG_ArgGly_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1042713 (http://www.snpedia.com/index.php/rs1042713)	
	4	ADRB2_CG_GlnGlu_gene_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1042714 (http://www.snpedia.com/index.php/rs1042714)	
	5	ADRB3_CT_ArgTrp_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4994 (http://www.snpedia.com/index.php/rs4994)	
	6	APOA2_265TC_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs5082 (http://www.snpedia.com/index.php/rs5082)	
	7	ACE_CG_ID_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4341 (http://www.snpedia.com/index.php/rs4341)	
	8	AGTmet_thr_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs699 (http://www.snpedia.com/index.php/rs699)	
	9	ACTN3_CT_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1815739 (http://www.snpedia.com/index.php/rs1815739)	
	10	BDKRB2_99_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1799722 (http://www.snpedia.com/index.php/rs1799722)	
	11	COL1A1_GT_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1800012 (http://www.snpedia.com/index.php/rs1800012)	
	12	COL5A1_CT_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs12722 (http://www.snpedia.com/index.php/rs12722)	
	13	CRP_1082GA_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1205 (http://www.snpedia.com/index.php/rs1205)	
	14	CYP1A2_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs762551 (http://www.snpedia.com/index.php/rs762551)	
	15	CAT_C262T_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1001179 (http://www.snpedia.com/index.php/rs1001179)	
	16	CETP_G279A_result	N/U					
	17	EPHX1_Tyr113His_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1051740 (http://www.snpedia.com/index.php/rs1051740)	
	18	FABP2_GA_AlaThr_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	Rs1799883 (http://www.snpedia.com/index.php/Rs1799883)	
	19	FTO_TA_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs9939609 (http://www.snpedia.com/index.php/rs9939609)	
	20	GDF5_CT_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs143383 (http://www.snpedia.com/index.php/rs143383)	
	21	GPX_Pro198Leu_result	YES	string		Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1050450 (http://www.snpedia.com/index.php/rs1050450)	

22	GSTM1_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	GSTM1deletion (http://www.snpedia.com/index.php/GSTM1)	
23	GSTT1_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	GSTT1deletion (http://www.snpedia.com/index.php/GSTT1)	
24	IL6_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1800795 (http://www.snpedia.com/index.php/rs1800795)	
25	IL6R_AC_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs2228145 (http://www.snpedia.com/index.php/rs2228145)	
26	LPL_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs328 (http://www.snpedia.com/index.php/rs328)	
27	LCT_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4988235 (http://www.snpedia.com/index.php/rs4988235)	
28	MTHFR_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1801133 (http://www.snpedia.com/index.php/rs1801133)	
29	NRF_AG_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs7181866 (http://www.snpedia.com/index.php/rs7181866)	
30	NOS3_G894T_result	N/U				
31	PPARG_CG_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1801282 (http://www.snpedia.com/index.php/rs1801282)	
32	PPARA_GC_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4253778 (http://www.snpedia.com/index.php/rs4253778)	
33	PPARGC1A_GA_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs8192678 (http://www.snpedia.com/index.php/rs8192678)	
34	TCF7L2_CT_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs7903146 (http://www.snpedia.com/index.php/rs7903146)	
35	TNF_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1800629 (http://www.snpedia.com/index.php/rs1800629)	
36	TRHR_TG_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs16892496 (http://www.snpedia.com/index.php/rs16892496)	
37	SOD2_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4880 (http://www.snpedia.com/index.php/rs4880)	
38	VDR_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs1544410 (http://www.snpedia.com/index.php/rs1544410)	
39	VEGF_CG_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs2010963 (http://www.snpedia.com/index.php/rs2010963)	
40	rs2395182_DQA1201_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs2395182 (http://www.snpedia.com/index.php/rs2395182)	
41	rs7775228_DQB1202_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs7775228 (http://www.snpedia.com/index.php/rs7775228)	
42	rs4713586_DQ4_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4713586 (http://www.snpedia.com/index.php/rs4713586)	
43	rs2187668_DQ25_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs2187668 (http://www.snpedia.com/index.php/rs2187668)	
44	rs4639334_DQ7_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs4639334 (http://www.snpedia.com/index.php/rs4639334)	
45	rs7454108_DQ8_result	YES	string	Genotype e.g. AG,CC,CG etc or ND if no data is available	rs7454108 (http://www.snpedia.com/index.php/rs7454108)	

Output Variables

CONCEPT		NAME	UNITS	POSSIBLE VALUES	DESCRIPTION
Overview	1	**Increase_Folate**		Folic Acid, Vit B6 and B12	Indicates whether you should increase Folic Acid, Vit B6 and B12 intake relative to the official RDA guidelines
	2	**Increase_Antiox**		Antioxidants	Indicates whether you should increase antioxidants intake relative to the official RDA guidelines
	3	**Increase_VitD**		Vitamin D	Indicates whether you should increase Vitamin D intake relative to the official RDA guidelines
	4	**Increase_Calcium**		Calcium	Indicates whether you should increase Calcium intake relative to the official RDA guidelines
	5	**Increase_Cruciferous**		Cruciferous	Indicates whether you should eat more cruciferous vegetables relative to the official RDA guidelines
	6	**Increase_Omega3**		Omega 3	Indicates whether you should increase Omega 3 intake relative to the official RDA guidelines
	7	**Increase_Fibre**		Fibre	Indicates whether you should increase fibre intake relative to the official RDA guidelines
	8	**Increase_Olive_Oil**		Olive Oil	Indicates whether you should increase olive oil intake relative to the official RDA guidelines
	9	**Decrease_Salt**		Salt	Indicates whether you should decrease salt intake relative to the official RDA guidelines
	10	**Decrease_Caffeine**		Caffeine	Indicates whether you should decrease caffeine relative to the official RDA guidelines
	11	**Decrease_SatFat**		Saturated Fats	Indicates whether you should decrease saturated fats relative to the official RDA guidelines
	12	**Decrease_CHO**		Refined carbs / sugars	Indicates whether you should decrease carbohydrate intake relative to the official RDA guidelines
	13	**Decrease_GrilledMeat**		Grilled Meat	Indicates whether you should eat less grilled meat relative to the official RDA guidelines
	14	**Lactose**		Lactose	Lactose intolerance
	15	**Celiac**		Negative or Possible predisposition	Possible predisposition to celiac disease
Genes		**INPUT.Genes**			
Gene_Impact	1	**GSTM1_Impact**		| * | **	Estimated gene impact
	2	**GSTT1_Impact**		| * | **	Estimated gene impact
	3	**TNF_Impact**		| * | **	Estimated gene impact
	4	**IL6_Impact**		| * | **	Estimated gene impact
	5	**MTHFR_Impact**		| * | **	Estimated gene impact
	6	**PPARG_Impact**		| * | **	Estimated gene impact
	7	**LPL_Impact**		| * | **	Estimated gene impact
	8	**AGTsalt_Impact**		| * | **	Estimated gene impact
	9	**ACEsalt_Impact**		| * | **	Estimated gene impact
	10	**ACEcarb_Impact**		| * | **	Estimated gene impact
	11	**SOD2_Impact**		| * | **	Estimated gene impact
	12	**APOC3_Impact**		| * | **	Estimated gene impact
	13	**VDR_Impact**		| * | **	Estimated gene impact
	14	**LCT_Impact**		| * | **	Estimated gene impact
	15	**ADH1C_Impact**		| * | **	Estimated gene impact

	16	CYP1A2caffeine_Impact		| * | **	Estimated gene impact
	17	CYP1A2meat_Impact		| * | **	Estimated gene impact
	18	TCF7L2_Impact		| * | **	Estimated gene impact
	19	ADRB2GlnGlu_Impact		| * | **	Estimated gene impact
	20	APOA2_impact		| * | **	Estimated gene impact
	21	FTO_impact		| * | **	Estimated gene impact
	22	FABP2_impact_impact		| * | **	Estimated gene impact
	23	CAT_impact		| * | **	Estimated gene impact
	24	EPHX1_impact		| * | **	Estimated gene impact
	25	GPX1_impact		| * | **	Estimated gene impact
	26	ADRB2_FAT		| * | **	Estimated gene impact
	27	ADRB3_FAT		| * | **	Estimated gene impact
Table_action	1	salt		Intermediate sensitivity to salt, <2,200 mg / day sodium| NAA| Normal sensitivity to salt, <2,400 mg / day sodium| Sensitive to salt, <2,200 / day sodium	Recommended action
	2	Alcohol		NAA| Positive effect of alcohol on cholesterol| Reduced beneficial effect of alcohol on cholesterol	Recommended action
	3	Olive_Oil		Increase olive oil intake| NAA| Standard recommendations for olive oil	Recommended action
	4	Caffeine		Limit caffeine intake|Standard recommendation	Recommended action
	5	Grilled_Meat		Limit intake of grilled meat|Standard recommendation	Recommended action
	6	Cruciferous		Consume 3-4 servings of cruciferous per week| Standard recommendation for cruciferous: 1-2 servings per week	Recommended action
	7	Inflammation		Normal: 1.6 g Omega 3 / day| Intermediate: 2 g Omega 3 / day| Increased basal inflammation: 3 g Omega 3 / day	Recommended action
	8	Sat_Fat		Limit saturated fat intake to < 16g / day| Standard recommendation for saturated fats	Recommended action
	9	Folate		Standard recommendations Vitamin B| Intermediate: at least 400 µg folic acid, 10 mg Vit B6, 15 µg Vit B12 per day|Slow: at least 600 µg folic acid, 15 mg Vit B6, 20 µg Vit B12 per day	Recommended action
	10	CHO		Limit intake of refined carbohydrates: glycemic load < 80 / day; consume at least 25 g/day fibre| Limit intake of refined carbohydrates: glycemic load < 70-80 / day; consume at least 25 g/day fibre |Limit intake of refined carbohydrates: glycemic load < 70 / day; consume at least 25 g/day fibre| Normal for refined carbohydrates: Glycemic load < 100/day	Recommended action
	11	Oxidative_stress		Standard recommendation for antioxidants; Increase antioxidants	Recommended action
	12	Selenium		Increase Selenium: 90 mcg / day|Increase Selenium: 105 mcg / day	Recommended action
	13	Nickel		Normal|Intermediate predisposition for nickel sensitivity|Increased predisposition for nickel sensitivity	Recommended action
	14	VitD		Normal: 600 IU / day Vitamin D|Increase: 800 IU / day Vitamin D	Recommended action
	15	Lactose		Lactose tolerant|Lactose intolerant	Recommended action
	16	Lactose_a		Reduce or avoid lactose|Normal for lactose	Recommended action
	17	Celiac		Very low celiac disease risk|Possible predisposition for celiac disease	Recommended action
	18	Celiac_a		Very low celiac disease risk|Possible predisposition for celiac disease	Recommended action
Goal_table	1	Vits_B		*	* = > To increase or decrease relative to the official RDA guidelines
Nutrient goal and limits	2	Antiox		*	* = > To increase or decrease relative to the official RDA guidelines
	3	Selenium		*	* = > To increase or decrease relative to the official RDA guidelines
	4	VitD		*	* = > To increase or decrease relative to the official RDA guidelines
	5	Fibra		*	* = > To increase or decrease relative to the official RDA guidelines
	6	Omega3		*	* = > To increase or decrease relative to the official RDA guidelines
	7	Calcium		*	* = > To increase or decrease relative to the official RDA guidelines
	8	Salt		*	* = > To increase or decrease relative to the official RDA guidelines
	9	PhysicalActivity		*	* = > To increase or decrease relative to the official RDA guidelines

pg. 23

	No	Name	Units	Values	Description
	10	Caffeine		*	* = > To increase or decrease relative to the official RDA guidelines
	11	SatFat		*	* = > To increase or decrease relative to the official RDA guidelines
	12	CHO		*	* = > To increase or decrease relative to the official RDA guidelines
	13	Folate_goal	µg	400|600|800	Folic acid
	14	B6_goal	mg	2|8|15	Vitamin B6
	15	B12_goal	µg	2|10|20	Vitamin B12
	16	VitA_goal		2,700 IU / 810 µg | 5,000 IU / 1500 µg	Vitamin A
	17	VitC_goal	mg	250|105	Vitamin C
	18	VitE_goal		200 IU / 180 mg | 15 IU / 13.5 mg	Vitamin E
	19	VitD_goal		800 IU / 20 µg | 600 IU / 5 µg	Vitamin D
	20	Fibra_goal	mg	25 | NA	Fibre
	21	Omega3_goal	g	1,6 | 2 | 3	Omega 3
	22	Calcium_goal	mg	1000 | 1300	Calcium
	23	Salt_goal	g	2,2 | 2,4	Salt
	24	Selenium_goal	µg	75 | 90 | 105 | ""	Selenium
	25	PhysicalActivity		45 min / day | 30 min / day	Physical activity
	26	Caffeine_goal	mg	300 | 200	Caffeine
	27	SatFat_goal	mg	16 | 22	Saturated fats
	28	CHO_goal		100 | 70 | 80 | NA	Max glycemic load / day
DietPanel					
ExerciseEER					
ExerciseCalories	1	EERtotal	kcal/day		Estimated Energy Requirement
	2	BMIround			Body Mass Index
Scores					
	1	FatPercent_round			Saturated Fats sensitivity score (based on genes) [1-10] e.g. 5.3
	2	CarbPercent_round			Refined carb sensitivity score (based on genes) [1-10] e.g. 8.2
	3	ExercisePercent_round			
Table_action	1	Refined_carbs_max		10% | 6% | 8% | NA	Refined carbohydrates: Recommended maximum percent of total calories
	2	Fibre_recc	g	25 | 30 | NA	Recommended daily consumption of fibre
	3	SatFat_max		10% | 6% | 8% | NA	Saturated fats: Recommended maximum percent of total calories
	4	Exercise_recc		increased | Moderately increased | medium | NA	Recommended exercise intensity level
	5	Exercise_recc_amount		high | medium | medium high | NA	Recommended exercise level of 30-45 mins 5 days per week with at least half from *Exercise_recc_amount* intensity activities
	6	Glicemic_load		100 | 70 | 80 | NA	Max glycemic load / day

pg. 24

C# Example

```csharp
using System;
using System.Collections.Generic;
using System.Net.Http;
using System.Text;
using System.Threading.Tasks;
using Newtonsoft.Json;
using Newtonsoft.Json.Linq;
using System.Xml.Serialization;
using System.Xml.Linq;
using System.Xml;
using System.IO;

namespace QualifyAPIClient
{
    public class Program
    {
        static void Main()
        {
            RunAsync();
            Console.ReadLine();
        }

        static async void RunAsync()
        {
            using (var client = new HttpClient())
            {
                client.BaseAddress = new Uri("http://api.quisper.eu/");
                client.DefaultRequestHeaders.Add("user_key", "YOUR_USER_KEY");

                // Create a new Case
                Case aCase = new Case();
                aCase.Name = "Qualify";
                // Create the Concept "Personal_Details" and fill in the required Variables
                Concept pd = new Concept();
                pd.Name = "Personal_Details";
                Variable Gender = new Variable("Gender", "M"); pd.Variables.Add(Gender);
                Variable Name = new Variable("Name", "SAMPLE101"); pd.Variables.Add(Name);
                Variable DoB = new Variable("DoB", "08/07/1975"); pd.Variables.Add(DoB);
                Variable Weight = new Variable("Weight", "81"); pd.Variables.Add(Weight);
                Variable Height = new Variable("Height", "1.83"); pd.Variables.Add(Height);
                Variable Age = new Variable("Age", "54"); pd.Variables.Add(Age);
                Variable Activity_level = new Variable("Activity_level", "low");
pd.Variables.Add(Activity_level);
                // Add the Concept to the Case
                aCase.Concepts.Add(pd);
                // Create the Concept "Genes" and fill in the required Variables
                Concept genes = new Concept();
                genes.Name = "Genes";
                Variable APOC3_result = new Variable("APOC3_result", "GG");
genes.Variables.Add(APOC3_result);
                Variable ADH1C_result = new Variable("ADH1C_result", "AG");
genes.Variables.Add(ADH1C_result);
                Variable ADRB2_AG_ArgGly_result = new Variable("ADRB2_AG_ArgGly_result", "GG");
genes.Variables.Add(ADRB2_AG_ArgGly_result);
                Variable ADRB2_CG_GlnGlu_gene_result = new Variable("ADRB2_CG_GlnGlu_gene_result", "CG");
genes.Variables.Add(ADRB2_CG_GlnGlu_gene_result);
                Variable ADRB3_CT_ArgTrp_result = new Variable("ADRB3_CT_ArgTrp_result", "CT");
genes.Variables.Add(ADRB3_CT_ArgTrp_result);
                Variable APOA2_265TC_result = new Variable("APOA2_265TC_result", "CC");
genes.Variables.Add(APOA2_265TC_result);
                Variable ACE_CG_ID_result = new Variable("ACE_CG_ID_result", "CG");
genes.Variables.Add(ACE_CG_ID_result);
                Variable AGTmet_thr_result = new Variable("AGTmet_thr_result", "TT");
genes.Variables.Add(AGTmet_thr_result);
                Variable ACTN3_CT_result = new Variable("ACTN3_CT_result", "TT");
genes.Variables.Add(ACTN3_CT_result);
                Variable BDKRB2_99_result = new Variable("BDKRB2_99_result", "CC");
genes.Variables.Add(BDKRB2_99_result);
                Variable COL1A1_GT_result = new Variable("COL1A1_GT_result", "GG");
genes.Variables.Add(COL1A1_GT_result);
                Variable COL5A1_CT_result = new Variable("COL5A1_CT_result", "TT");
genes.Variables.Add(COL5A1_CT_result);
                Variable CRP_1082GA_result = new Variable("CRP_1082GA_result", "CC");
genes.Variables.Add(CRP_1082GA_result);
                Variable CYP1A2_result = new Variable("CYP1A2_result", "AA");
genes.Variables.Add(CYP1A2_result);
                Variable CAT_C262T_result = new Variable("CAT_C262T_result", "AG");
genes.Variables.Add(CAT_C262T_result);
                Variable CETP_G279A_result = new Variable("CETP_G279A_result", "ND");
```

```csharp
                genes.Variables.Add(CETP_G279A_result);
                Variable EPHX1_Tyr113His_result = new Variable("EPHX1_Tyr113His_result", "CT");
                genes.Variables.Add(EPHX1_Tyr113His_result);
                Variable FABP2_GA_AlaThr_result = new Variable("FABP2_GA_AlaThr_result", "CC");
                genes.Variables.Add(FABP2_GA_AlaThr_result);
                Variable FTO_TA_result = new Variable("FTO_TA_result", "TT");
                genes.Variables.Add(FTO_TA_result);
                Variable GDF5_CT_result = new Variable("GDF5_CT_result", "CT");
                genes.Variables.Add(GDF5_CT_result);
                Variable GPX_Pro198Leu_result = new Variable("GPX_Pro198Leu_result", "CC");
                genes.Variables.Add(GPX_Pro198Leu_result);
                Variable GSTM1_result = new Variable("GSTM1_result", "D");
                genes.Variables.Add(GSTM1_result);
                Variable GSTT1_result = new Variable("GSTT1_result", "I");
                genes.Variables.Add(GSTT1_result);
                Variable IL6_result = new Variable("IL6_result", "CG"); genes.Variables.Add(IL6_result);
                Variable IL6R_AC_result = new Variable("IL6R_AC_result", "AA");
                genes.Variables.Add(IL6R_AC_result);
                Variable LPL_result = new Variable("LPL_result", "CC"); genes.Variables.Add(LPL_result);
                Variable LCT_result = new Variable("LCT_result", "CT"); genes.Variables.Add(LCT_result);
                Variable MTHFR_result = new Variable("MTHFR_result", "CT");
                genes.Variables.Add(MTHFR_result);
                Variable NRF_AG_result = new Variable("NRF_AG_result", "AA");
                genes.Variables.Add(NRF_AG_result);
                Variable NOS3_G894T_result = new Variable("NOS3_G894T_result", "ND");
                genes.Variables.Add(NOS3_G894T_result);
                Variable PPARG_CG_result = new Variable("PPARG_CG_result", "CC");
                genes.Variables.Add(PPARG_CG_result);
                Variable PPARA_GC_result = new Variable("PPARA_GC_result", "CG");
                genes.Variables.Add(PPARA_GC_result);
                Variable PPARGC1A_GA_result = new Variable("PPARGC1A_GA_result", "AG");
                genes.Variables.Add(PPARGC1A_GA_result);
                Variable TCF7L2_CT_result = new Variable("TCF7L2_CT_result", "TT");
                genes.Variables.Add(TCF7L2_CT_result);
                Variable TNF_result = new Variable("TNF_result", "AG"); genes.Variables.Add(TNF_result);
                Variable TRHR_TG_result = new Variable("TRHR_TG_result", "AA");
                genes.Variables.Add(TRHR_TG_result);
                Variable SOD2_result = new Variable("SOD2_result", "TT"); genes.Variables.Add(SOD2_result);
                Variable VDR_result = new Variable("VDR_result", "TT"); genes.Variables.Add(VDR_result);
                Variable VEGF_CG_result = new Variable("VEGF_CG_result", "CG");
                genes.Variables.Add(VEGF_CG_result);
                Variable rs2395182_DQA1201_result = new Variable("rs2395182_DQA1201_result", "TT");
                genes.Variables.Add(rs2395182_DQA1201_result);
                Variable rs7775228_DQB1202_result = new Variable("rs7775228_DQB1202_result", "TT");
                genes.Variables.Add(rs7775228_DQB1202_result);
                Variable rs4713586_DQ4_result = new Variable("rs4713586_DQ4_result", "TT");
                genes.Variables.Add(rs4713586_DQ4_result);
                Variable rs2187668_DQ25_result = new Variable("rs2187668_DQ25_result", "GG");
                genes.Variables.Add(rs2187668_DQ25_result);
                Variable rs4639334_DQ7_result = new Variable("rs4639334_DQ7_result", "AA");
                genes.Variables.Add(rs4639334_DQ7_result);
                Variable rs7454108_DQ8_result = new Variable("rs7454108_DQ8_result", "TT");
                genes.Variables.Add(rs7454108_DQ8_result);
                // Add the Concept "Genes" to the Case
                aCase.Concepts.Add(genes);

                string requestUri = "safecape/products/qualify/execute";
                var response = client.PostAsync(requestUri,
                    new StringContent(JsonConvert.SerializeObject(aCase).ToString(),
                        Encoding.UTF8, "application/json"))
                    .Result;
                if (response.IsSuccessStatusCode)
                {
                    Console.WriteLine(string.Format("Resource: {0}", requestUri));
                    Console.WriteLine(string.Format("HTTP Status Code: {0}{1}", response.StatusCode,
Environment.NewLine));
                    dynamic content =
JsonConvert.DeserializeObject(response.Content.ReadAsStringAsync().Result);
                    string salt = content.ModelOutput.Concepts[3].Variables[0].Value;
                    Console.WriteLine(string.Format("Your sensitivity to salt: {0}", salt));
                    string caffeine = content.ModelOutput.Concepts[3].Variables[3].Value;
                    Console.WriteLine(string.Format("What about caffeine?: {0}", caffeine));
                    string lactose = content.ModelOutput.Concepts[3].Variables[14].Value;
                    Console.Write(string.Format("You are: {0}", lactose));
                    Console.WriteLine();
                    Console.WriteLine();
                }
                else
                {
                    Console.WriteLine(string.Format("HTTP Status Code: {0} {1}", response.StatusCode,
response.Content.ReadAsStringAsync().Result));
                }

            }
        }
```

```csharp
#region Rules Toolset Model

public class Variable
{
    [XmlAttribute("name")]
    public string Name { get; set; }
    [XmlAttribute("value")]
    public string Value { get; set; }

    public Variable()
    {

    }

    public Variable(string name, string value)
    {
        this.Name = name;
        this.Value = value;
    }
}

public class Concept
{
    [XmlAttribute("name")]
    public string Name { get; set; }

    [XmlElement("variable")]
    public List<Variable> Variables { get; set; }

    [XmlElement("concept")]
    public List<Concept> Concepts { get; set; }

    public Concept()
    {
        Variables = new List<Variable>();
        Concepts = new List<Concept>();
    }
}

[XmlRoot("case")]
public class Case
{
    public Case()
    {
        Concepts = new List<Concept>();
    }

    [XmlAttribute("name")]
    public string Name { get; set; }

    public string Use { get; set; }

    [XmlElement("concept")]
    public List<Concept> Concepts { get; set; }

    public string Serialize()
    {
        XmlSerializer x = new XmlSerializer(typeof(Case));
        XDocument doc = new XDocument();
        using (XmlWriter xw = doc.CreateWriter())
        {
            x.Serialize(xw, this);
            xw.Close();
        }
        XElement el = doc.Root;
        return el.ToString();
    }

    public Case Deserialize(XElement xElement)
    {
        Case quoter = null;
        XmlSerializer x = new XmlSerializer(typeof(Case));
        using (XmlReader xr = xElement.CreateReader())
        {
            quoter = x.Deserialize(xr) as Case;
            xr.Close();
        }
        return quoter;
    }

    public Case Deserialize(string xml)
    {
        Case result = null;
        XmlSerializer x = new XmlSerializer(typeof(Case));
```

```csharp
            using (TextReader reader = new StringReader(xml))
            {
                result = (Case)x.Deserialize(reader);
            }
            return result;
        }
    }

    #endregion

}
```