# GeekUp Challenge II

This challenge is a fun activity which aims to gain more team work experience as well as improving in agile practices in engineering.

In this challenge each team, will attach themselves to a webservice (details below). They keep receive GET requests from the service. They have to modify their code to answer the question(s) they receive from the server and respond.

Teams score when they answer correctly (ie +10) they will have negative points if they don’t respond (ie -20) and fewer negative point if they answer wrong. Points for different questions are different.

Because the questions change so frequent, and still previous question patterns appear, it is highly recommended to use unit test to check your piece of code.

Also it is highly recommended to use refactoring practices to have improved your code to cover more questions efficiently.

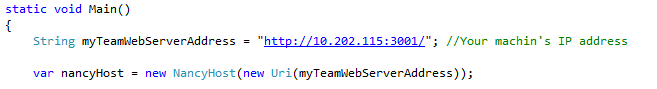
Focus on your idea, build, release, learn and iterate them.

We try to time box it to around 2 hours, depends how it goes. We will start with warmup mode, which the question is what your team name is. When everyone managed to attach itself to the service and had correct response, we restart the service with actual questions.

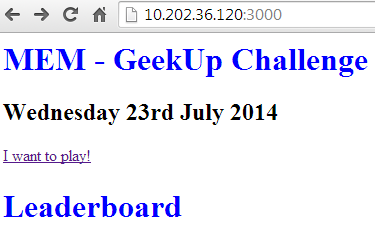
Questions are usually basic maths questions, but in text format. You need to parse them and response with correct answer. The questions repeat with different values. When you receive new patterns of questions, you still receive the basic ones.

## Getting started

* Install NuGet (from [**http://nuget.org/**](http://nuget.org/))
* Open the Solution in Visual Studio 2010
* To get Nancy
  + Go to Tool | Library Package Manager | Package Manager Console
  + Run: Install-Package Nancy.Hosting.Self
* Download NancySelfHosting project from \\Icc1\Shared\GeekupChallenge
* Modify the webserver address to your machine IP address in Program.cs



* Run the program!
* Go to <http://10.202.36.120:3000>



* Attach yourself to the game by clicking I want to play!
* Add your team name and your simple webserver URL : [your machine IP address]:3001



* You are attached!

**http://[Your machine IP]:3001**

HTTP response

**Name:** Alice

**URL:** [Your machine IP]:3001

HTTP GET /?q=

Very simple web server

Server

http://10.202.36.120:3000/

Player

## How to play

* The server frequently sends a get request to your simple webserver
* You HTTP response should contain the answer
* You can change the answer value in Module.cs

public class Question

{

public string questionType;

public string question;

public Object parameters;

public int points;

}

public class Module : NancyModule

{

public Module()

{

Get["/"] = parameters => { return handleQuery(parameters); }; }

private String handleQuery(dynamic parameters)

{

String query = "";

query += HttpUtility.UrlDecode(Request.Url.Query);

if (query.Length > 0)

{

query = query.Substring(13, query.Length - 13);

DataContractJsonSerializer js = new DataContractJsonSerializer(typeof(Question));

MemoryStream stream = new MemoryStream(Encoding.UTF8.GetBytes(query));

Question q = (Question)js.ReadObject(stream);

Debug.WriteLine(q.question);

}

String answer = "Hoss";

return answer;

}

}

* In WARMUP mode, the service needs to responds the team name
* If you respond with correct answer, you gain 10 points
* If no response or error, -20
* For wrong answers fewer points
* When you attach, server provides you with a URL where you can monitor your results



## Rules

* Every 10 minutes the level gears up
* At any moment of time, the questions are from one to 4 categories
* Question from wach category has its own point
* Just play fairly!

## Some links

<https://console.developers.google.com/project/apps~sapient-notch-530/apiui/api/translate/method/language.translations.list?authuser=0>

<https://code.google.com/p/currency-converter-api/>

<http://rubydoc.info/github/RubyMoney/google_currency/master/frames>

<https://developer.forecast.io/docs/v2>

<http://rate-exchange.appspot.com/currency?from=USD&to=EUR&q=1>